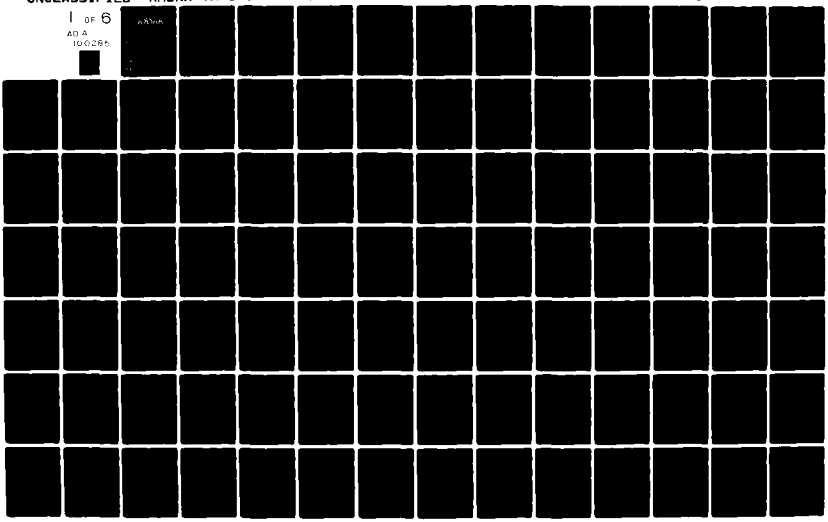
AD-A100 285 ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY ABERDEEN PROV--ETC F/G 19/1
COPPERHEAD OPERATIONAL PERFORMANCE EVALUATION (COPE): COMPUTER --ETC(U)
MAR 81 R S SANDMEYER
UNCLASSIFIED AMSAA-TR-318 NL

1 OF 6
AD A
100285



LEVEL II

AD

# AMSAA

12

TECHNICAL REPORT NO. 318

AD A100285

# COPPERHEAD OPERATIONAL PERFORMANCE EVALUATION (COPE): COMPUTER PROGRAM USER AND ANALYST MANUAL

DTIC

RICHARD S. SANDMEYER

MARCH 1981

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

DTIC FILE COPY

U. S. ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY
ABERDEEN PROVING GROUND, MARYLAND

81 6 15 054

## DISPOSITION

Destroy this report when no longer needed. Do not return it to the originator.

## DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position unless so specified by other official documentation.

## WARNING

Information and data contained in this document are based on the input available at the time of preparation. The results may be subject to change and should not be construed as representing the DARCOM position unless so specified.

## TRADE NAMES

The use of trade names in this report does not constitute an official endorsement or approval of the use of such commercial hardware or software. The report may not be cited for purposes of advertisement.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

# REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

| | |
|---|---|
| 1. REPORT NUMBER: Technical Re[redacted]t, No. 318 | 2. GOVT ACCESSION NO.: AD-A100 285 |
| 3. RECIPIENT'S CATALOG NUMBER | |
| 4. TITLE (and Subtitle): COPPERHEAD Operational Performance Evaluation (COPE): Computer Program User and Analyst Manual | 5. TYPE OF REPORT & PERIOD COVERED |
| | 6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s): Richard S. Sandmeyer | 8. CONTRACT OR GRANT NUMBER(s): AMSAA-TR-318 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS: Director, US Army Materiel Systems Analysis Activity, Aberdeen Proving Ground, MD 21005 | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: DA Project # 1R765705M541 |
| 11. CONTROLLING OFFICE NAME AND ADDRESS: Commander, US Army Materiel Development & Readiness Command, 5001 Eisenhower Ave., Alexandria, VA 22333 | 12. REPORT DATE: Mar[redacted]81 |
| | 13. NUMBER OF PAGES: 490 (506) |
| 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) | 15. SECURITY CLASS. (of this report): UNCLASSIFIED |
| | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

COPPERHEAD
COPPERHEAD Operational Performance Evaluation
Combat Operational Performance Evaluation
COPE

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The COPE model simulates a COPPERHEAD fire mission under battlefield conditions. It takes into account such degrading effects on designator, fire direction center (FDC), artillery battery, and COPPERHEAD round performance as meteorological conditions, terrain shielding, smoke, dust, enemy artillery, preparatory fires, enemy direct fire against the designator, communication failures, and target location error as well as numerous factors that affect COPPERHEAD's ability to engage, hit, and kill a target.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19. Key Words - Continued

XM712
Artillery
Weapons Effectiveness
Item Level Performance
Battlefield Environment
Operational Performance
Laser Guided Weapons
Smart Weapons

20. Abstract - Continued

This report includes detailed documentation of the computer programs that implement the COPE model. Sample cases complete with input and output as well as program source listings are included.

TABLE OF CONTENTS

TABLE OF CONTENTS - Continued

TABLE OF CONTENTS - Continued

TABLE OF CONTENTS - Continued

TABLE OF CONTENTS - Continued

TABLE OF CONTENTS - Continued

TABLE OF CONTENTS - Continued

PAGE

TABLE OF CONTENTS - Continued

TABLE OF CONTENTS - Continued

## LIST OF COMPUTER LISTINGS

## LIST OF FIGURES

LIST OF TABLES



Next page is blank.

# PREFACE

This report documents the COPE methodology and computer programs.

Various sections of this report are aimed at different audiences:

Chapter 1 is an introduction to COPE and to this report.

Chapters 2 and 3 are intended for the analyst who is concerned with the modeling methodology and assumptions.

Chapters 4, 5, 6, 7, 14, 15, and 16 are aimed at the user who wants to use the COPE model and its preprocessors in production runs, but who is not interested in programming details.

Chapters 8, 9, 10, 11, 12, and 13 are intended for the programmer who wants a thorough understanding of the COPE computer code in order to implement it on other machines or simply to make changes to it.

The first four appendixes (A, B, C, and D) include program listings and sample cases. The sample cases are cumulative in the sense that the three PAM sample input sets should be run using respectively the three PAM sample runstreams before running the PREPMS program. The sample PREPMS case should then be run using the TAPE 11 created by the PAM runs (this will result in the output of Appendix C). Finally, the COPE sample case should be run using the TAPE 11 created by PAM and PREPMS (this should result in the output of Appendix D). Running these sample cases and comparing the results with those in the appendixes provide a check for correct implementation of COPE and its preprocessors on other machines.

If discrepancies are found in this publication or the associated software, please send notice to:

Director<br>
US Army Materiel Systems Analysis Activity<br>
ATTN: DRXSY-GS (Mr. R. Sandmeyer)<br>
Aberdeen Proving Ground, MD 21005

Suggestions for improvements to this report or the associated programs may also be sent to the above address.

In addition, requests for magnetic tape copies of the COPE programs and related files should be sent to the same address along with the appropriate paperwork for such requests.

This report may be updated by the issuing of revised pages. The frequency of such updates will, of course, depend on the number of errors requiring correction, the changes made in the model, and the demand for the report. Those desiring to maintain the latest versions of the report and the programs should be certain that they are on the COPE distribution list to be maintained by AMSAA.

**Next page is blank.**

## NOTE ON CLASSIFICATION

This report is unclassified, however, when the COPE model is used for actual production runs, it requires some classified inputs. The inputs used in the sample cases (see Appendixes) are correct data except for false values which have been inserted in place of those data whose correct values are classified.

In addition, comment cards with the characters "CSEC" in columns 1-4 point out places in the program code where fictitious constants have been substituted to replace classified values in the listings of PAM and COPE (Listings A-1 and D-1).

The following data are classified (at the time of publication of this report) when correct values are used and identified as such:

In the PAM program: ETH, THEMH, and DISMH are classified by Reference 1 below. ED is classified by Reference 2.

In PREPMS and COPE programs, the values of PE, RNGTTF, and PKTBL, as well as the true seeker sensitivity value, are all classified by Reference 1.

---

[1]Security Classification Guide, Projectile, 155mm, Cannon Launched, Guided, XM712, (CLGP), dated 24 July 1979.
[2]Security Classification Guide, Ground Laser Designators, dated 16 June 1978.

Next page is blank.

ACKNOWLEDGEMENTS

The COPE model was developed by the Support Weapons Analysis Branch (SWAB) of the Ground Warfare Division (GWD) of the US Army Materiel Systems Analysis Activity (AMSAA).

The requirement to create such a model came from David Hardison, Deputy Under Secretary of the Army for Operations Research (DUSA-OR). A preliminary model proposal was put forward by Herbert Fallin of the Office of DUSA-OR.

When AMSAA received the requirement and proposed model, a team of four analysts of the System Evaluation Section of SWAB was established to develop the model, oversee the collection of data, and, as it turned out, participate in the COPPERHEAD COEA by doing production runs and writing up the results. This team consisted of the section chief, Richard Scungio, who served as team leader and overall coordinator, Julian Chernick, Michael Starks, and the author of the present report, Richard Sandmeyer. Much of the design of the COPE model was accomplished by discussion within the team and individual credit cannot easily be assigned. However, COPE drew heavily on the probability of engagement work done previously by Mr. Chernick and Mr. Starks (see bibliography) including the modeling of weather; the PAM program written by Michael Starks was a derivative of a program used in that earlier work. Mr. Sandmeyer converted the conceptual COPE model into a working computer program (COPE) and also wrote the remaining two programs of this report (PREPMS and PRBLOS).

This report is, of course, concerned with the COPE model and its implementation as a computer program, but a word of appreciation to those who helped gather data for COPE is appropriate here even though this report does not deal with the data collection process (for the data sources used, see <u>COPPERHEAD COEA</u> in bibliography). Data for use in COPE was gathered by David Barnhart, Diana Frederick, Annie Young, and Edward Stauch (all of SWAB). In addition, David Kilminster of SWAB had previously assisted Mr. Chernick and Mr. Starks in obtaining and using weather data. Charles Cairns of AMSAA supplied LDWSS probability of hit numbers and MAJ Charles Williams of FT Sill, OK, provided advice and served as liaison between AMSAA and FT Sill. The SMOKE data was obtained from Sid Gerard (then of AMSAA, now of PM, SMOKE).

Special thanks is given to Mrs. Robin DeFranks, who edited and typed this lengthy report. Thanks also to Mary Webb of SWAB who typed the text in the flow charts of Chapter 11.

Next page is blank.

CHAPTER 1

# 1. INTRODUCTION

## 1.1 COPE Background.

Traditionally the item level performance of a weapon system has been expressed in terms of the firepower characteristics of the weapon. The weapon characteristics normally included in the computation of firepower were range, accuracy, lethality, and rate-of-fire. With the introduction of more sophisticated weapons, an awareness of the need to include additional factors in the assessment of a weapon's performance has emerged at both the decision and R&D levels. These additional factors have collectively been referred to as the "Battlefield Environment" and include natural phenomena as well as conditions imposed on the system by friendly and enemy forces.

The need for quantifying a weapon system's performance within the battlefield environment recently became most critical with the development of the 155mm XM712 COPPERHEAD artillery projectile. To study its potentially degraded performance under some battlefield environments, AMSAA developed a computer model which is known by the acronym COPE (COPPERHEAD Operational Performance Evaluation).

In addition to being a model for assessing the performance of a weapon system, the model and the manner of generating the input requirements constitute a novel approach to the evaluation of weapon systems. This approach results in a manner of assessing the performance of most weapon systems and potentially can quantitatively produce insight into the performance of one or more weapon systems which previously could only be treated subjectively.

This report describes the COPE model as currently used at AMSAA for the assessment of COPPERHEAD. AMSAA has modified the computer code to obtain models that analyze the performance of other weapon systems within the context of a COPE type "operational performance" analysis. The term COPE itself has now come to have two meanings: (1) "Combat Operational Performance Evaluation" which is the generic name for all of the derivative models of the original COPPERHEAD version and (2) "COPPERHEAD Operational Performance Evaluation" which is the first model in this family of "Operational Performance" models and is the one described in this report.

The models in the COPE family include:

(1) COPPERHEAD Operational Performance Evaluation (COPE) used to evaluate the XM712 COPPERHEAD artillery round,

(2) Tank Performance Evaluator (TAPE) used as part of the XM1 tank evaluation,

(3) TOW Operational Performance Evaluator (TOPE) used to evaluate the TOW anti-tank missile,

(4) SADARM Operational Performance Evaluator (SOPE) used to evaluate the proposed SADARM artillery round,

(5) Top Attack Evaluation Model (TAEM) used to evaluate candidates in the Top Attack Study,

(6) Ground Launched HELLFIRE Operational Performance Evaluation (GLHOPE) used to evaluate ground launched HELLFIRE,

(7) RPVCOPE used to evaluate COPPERHEAD when designation was performed by an RPV, and,

(8) HELLFIRE Operational Performance Evaluation (HOPE) used to evaluate HELLFIRE launched from helicopters.

The input requirements for these models range from moderate to massive. The TAPE and TOPE models for example require no preprocessors and take less input than the COPPERHEAD version. The TAEM program, on the other hand, requires as input results generated by the very costly, long running, TGSM (Terminally Guided Submunition Model) program.

This report is concerned only with the COPE model (in all remaining sections of this report, COPE shall have its second, narrower meaning: "COPPERHEAD Operational Performance Evaluation"); however, much of what is discussed applies to the other models in the family. Any of the models of the COPE family can be obtained from AMSAA, though at the present time only the COPPERHEAD and tank versions have been extensively documented.

## 1.2 Overview of COPE Computer Programs.

There are actually four computer programs used in the COPPERHEAD Operational Performance Evaluation. They are:

(1) The main COPE program (formerly called SLATCH for Simple Look at the COPPERHEAD) descended from the original proposal put forward by DUSA-OR.

(2) The PAM (Probability of Seeker Acquisition and Round Maneuver) model used to create the probability of engagement tables used as input to the main COPE program. This model is described and documented more thoroughly in a forthcoming AMSAA technical report.

(3) The PREPMS (Preprocessor Mass Storage) program that is used to create and modify the word addressable mass storage file used as the COPE "data base".

(4) The PRBLOS (Probability of Line-of-Sight) model used to obtain the probability of having line-of-sight for the terminal phase of the COPPERHEAD trajectory.

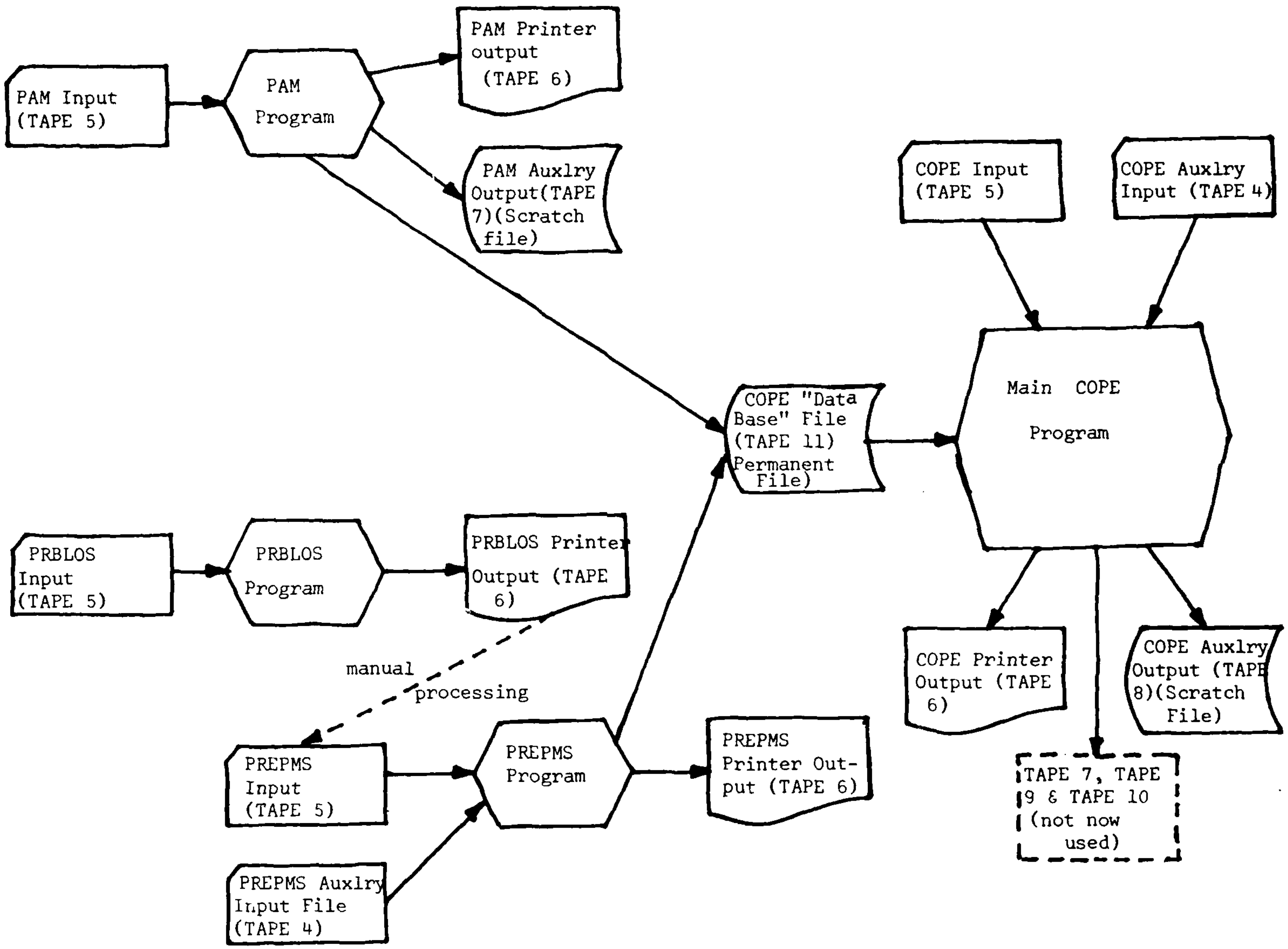



FIGURE 1-1 COPE And Its Preprocessors

All four of these programs are documented in this report, though the PAM model is covered in much less detail than the other three since a more detailed treatment of it will be published in a separate report.

The relationships among the four programs and the word addressable "data base" file are shown in Figure 1-1.

The major data sets (called "data blocks" later in this report) are put on the data base by PAM and PREPMS. Then when one wishes to run COPE, it is necessary only to submit a string of simple input commands describing the cases to be run. For example, when running a case using December weather for 0600 (6 a.m.), one need merely submit a card: WEATHER, DECEMBER, 0600 rather than read in thirty or forty cards filled with the appropriate weather data. This arrangement makes it extremely simple to run many cases of COPE with relatively few input cards once the data base file has been created.

In the event that one wishes to run cases that use a data set not on the data base file, there is a "temporary" option that allows one to read the data into COPE directly; however, if one intends to run multiple cases with such data, it will be easier in the long run to read it onto the permanent data base file once using the PREPMS program and then "call it up" for use as necessary in COPE by submitting the appropriate single card command.

All four programs are written in CDC FORTRAN Extended Version 4.

# CHAPTER 2

## 2. DESCRIPTION OF COPPERHEAD MISSION MODELING

### 2.1 Application Area of Simulation.

The main COPE program is a simulation of a COPPERHEAD fire mission under battlefield conditions. It takes into account such degrading effects on designator, fire direction center (FDC), artillery battery, and COPPERHEAD round performance as meteorological conditions, terrain shielding, smoke, dust, enemy artillery preparatory fires, enemy direct fire against the designator, communication failures, and target location error as well as numerous factors that affect COPPERHEAD's ability to engage and hit a target.

The main COPE program was designed to evaluate the operational performance of the overall COPPERHEAD system under battlefield conditions.

The output of the program enables one to see which factors contribute most to the degrading of COPPERHEAD system performance and which are, therefore, areas in which performance improvement efforts should be concentrated.

In addition, by varying the input value(s) associated with a particular factor and then studying the results, one can perform the customary sensitivity analysis. A series of such analyses can be expected to show that changes (within realistic limits) in some factors have little effect on overall system operational performance whereas changes in other factors will result in major performance improvement or degradation.

By comparing those factors to which system operational effectiveness is most sensitive with those factors over which system designers and tacticians have some control, one can determine what, if any, design changes or employment doctrine changes might be made to improve overall COPPERHEAD system operational performance. This comparison and analysis, of course, is done outside the COPE program, but the resulting modified COPPERHEAD system and employment doctrine might then be evaluated in the COPE model or a descendent of it.

Finally, several derivative models have been developed from an earlier COPE program (SLATCH version 3.2) and are in use for evaluating tanks, TOW, and Hellfire under battlefield conditions. These models allow limited but informative comparisons of the operational performance of the different weapon systems.

## 2.2 Description of Model.

2.2.1. Outline of COPPERHEAD Mission. The heart of the main COPE program is the Monte Carlo simulation of a potential COPPERHEAD fire mission. This simulation consists of a series of steps that would occur in an actual COPPERHEAD fire mission. For a successful mission, these steps start with the unmasking from terrain shielding of a target and end with the killing of a target by COPPERHEAD. For an unsuccessful mission, the steps end before the target is killed.

The steps are grouped into three series bounded by four major events. The following outline gives the four major events as well as the steps (or tests) that must be passed to reach the next major event. All steps and events occur in the order listed.

This describes the tests when the so-called "shooting gallery" line-of-sight model is used. Following the list is a description of the modifications required when the "random occurrence" line-of-sight model is used.

Event 1. A target (one or more target vehicles) unmasks (i.e., enters a region within line-of-sight of the designator operator's position). This is termed an "occasion" and begins the simulation of the potential COPPERHEAD mission.

Test 1: Has the designator operator survived enemy preparatory artillery fires up to the present time?

Test 2: Is the target within the meteorological visibility range limit of the designator operator for the present weather conditions?

Test 3: Is the target within the designation range of the laser designator system being played?

Test 4: Did the designator operator detect/acquire the target before it again became masked by terrain?

Test 5: Is the line-of-sight from designator-to-target sufficiently free of smoke that the designator operator can see and designate the target?

Test 6: Is the line-of-sight from designator-to-target sufficiently free from dust for the designator operator (D.O.) to see and designate the target?

Test 7: Is the target sufficiently far away from the designator operator that he will stay and designate rather than "bail-out?"

Event 2: The designator operator has acquired a target and decides to engage it (i.e., call for COPPERHEAD fire against it). This is termed an "attemped engagement."

Test 8: Is the designator operator able to communicate with the fire direction center (FDC)?

Test 9: Does the message which the designator operator sends to the fire direction center contain the correct information and does the FDC interpret it correctly?

Test 10: Is the target still (after time has been spent on communication and preparing to fire the mission) sufficiently far away that the designator operator will stay and designate rather than "bail-out?"

Test 11: Does the designator operator still have line-of-sight to the target?

If tests 8 through 11 are all passed, then conditions are such that a COPPERHEAD fire mission occurs. This leads to:

Event 3: The battery is ready to fire and the designator is ready to designate. A COPPERHEAD round is fired. This is termed a "shot".

Test 12: Does the designator operator still have line-of-sight to the target during the final critical time interval of the COPPERHEAD trajectory?

Test 13: Has the designator operator been warned to begin lasing the target or has he seen the first COPPERHEAD round of this fire mission already impact?

Test 14: Does the designator survive direct fire from the target being designated and is the line-of-sight to the target unobscured by direct fire rounds landing in front of the designator?

Test 15: Is the COPPERHEAD round just fired reliable?

Test 16: Is the line-of-sight from the designator operator to the target vehicle unobscured by small terrain features?

Test 17: Does the round come sufficiently close to the target for the seeker to receive enough reflected laser energy to engage the target and is the target within the maneuver footprint of the COPPERHEAD?

Test 18: Does the COPPERHEAD round hit the target?

Test 19: Does the COPPERHEAD round kill the target?

If tests 12 through 19 are all passed, then we have:

Event 4: The target vehicle is killed. This is termed a "kill."

If more COPPERHEAD rounds are to be fired for this mission, the model time is moved ahead to account for the time between rounds, and the model resumes at Event 3 for the next COPPERHEAD round.

To model the "random occurrence" line-of-sight case, tests 4 and 11 are not made and test 12 uses a different assumption.

The two ways of handling the line-of-sight problem are called the "shooting gallery" method and the "random occurrence" method.

The "shooting gallery" method assumes that the entire COPPERHEAD mission must be completed during the time interval starting with the first target vehicle entering line-of-sight segment and ending with the last vehicle of the target leaving that same line-of-sight segment (see Figure 2-1).

The "random occurrence" method on the other hand assumes that target vehicles are passing in and out of view. Instead of assuming that all rounds fired after the last target vehicle leaves the line-of-sight segment in which it was acquired are wasted as in the "shooting gallery" technique, the "random occurrence" method does not check loss of line-of-sight until the last critical seconds of the COPPERHEAD trajectory and even then the check allows for the possibility that a target may have left the first line-of-sight segment but entered a second (or subsequent) line-of-sight segment where COPPERHEAD can be used. The "random occurrence" line-of-sight model, then merely checks whether there is at least one unkilled target vehicle that will be within line-of-sight of the designator for the final critical seconds of the COPPERHEAD trajectory (see Figure 2-2).

Now each step (test) in the model is explained in detail:

2.2.2 Preparatory Fires. The first test encountered by a potential COPPERHEAD mission is for designator operator survivability of preparatory fire. A probability that the designator operator is killed by enemy preparatory artillery is input (either explicitly or by default) and for each replication (i.e., each sample potential COPPERHEAD mission) of the given case, a uniformly distributed (pseudo-) random number is drawn. If that random number is less than the input probability that the designator operator is killed by enemy preparatory artillery fire, then the potential COPPERHEAD

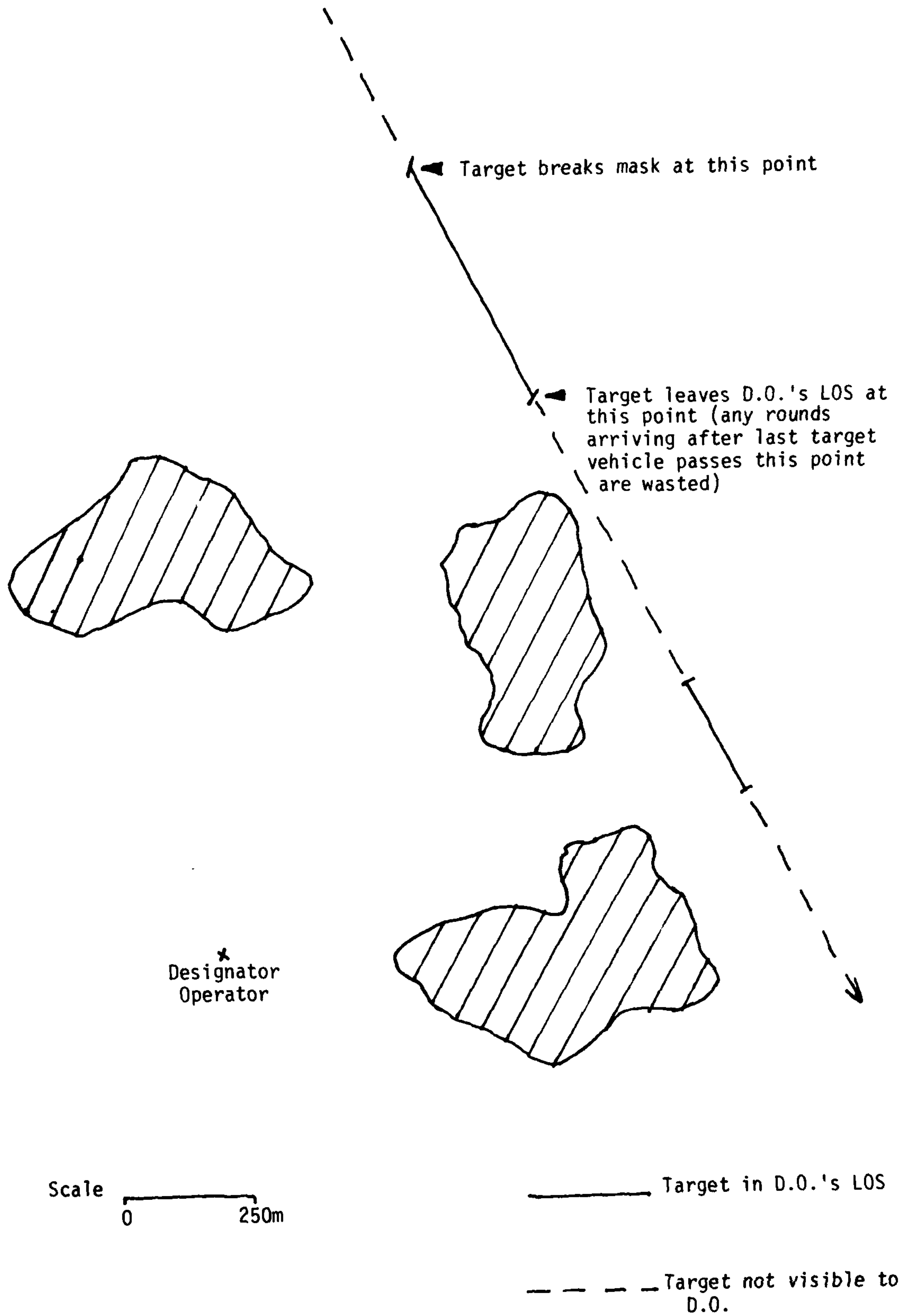



FIGURE 2-1 "Shooting Gallery" LOS Model

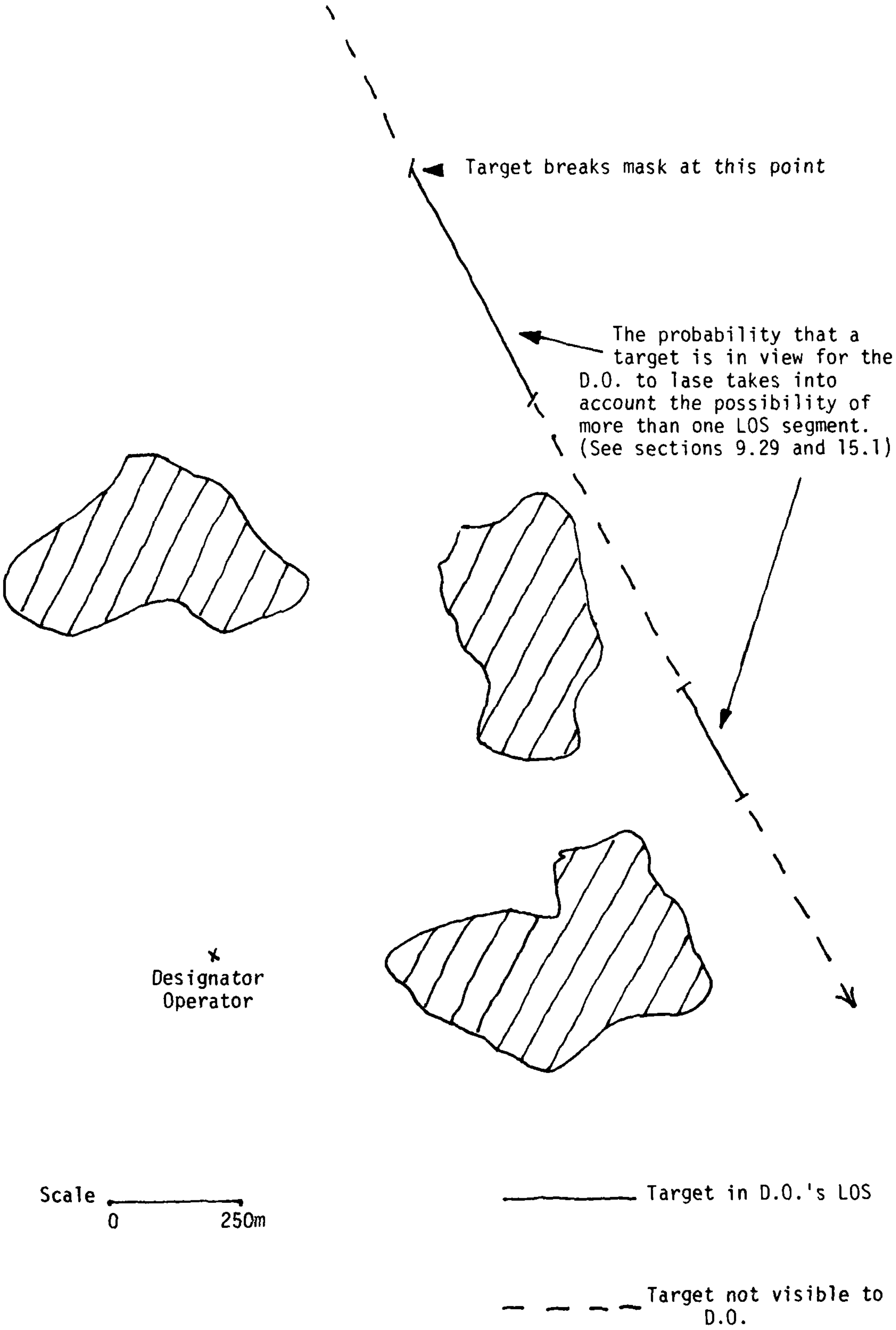



FIGURE 2-2 "Random Occurrence" LOS Model

mission aborts because the designator operator is killed; otherwise, the potential COPPERHEAD mission continues to the next test.

2.2.3 Visibility Range. The next test depends on the weather conditions. For the month and time of day that is used for a given case, there is a distribution of weather conditions which includes the probabilities of various cloud ceiling altitude and meteorological visibility range combinations as well as probabilities of various Pasquill atmospheric stability categories, windspeeds, and relative humidities. For each replication of a given case, the cloud ceiling altitude and meteorological visibility range distribution is sampled and the results (cloud ceiling altitude and meteorological visibility limit (range)) obtained are recorded for later use.

The program then samples the acquisition range distribution once for each replication to obtain the range from designator-operator (D.O.) to target terrain unmask for the current replication. If the target unmasks from terrain shielding at a range from the designator-operator beyond the current replication's meteorological visibility limit, then the potential COPPERHEAD mission is aborted since the target cannot be seen and, therefore, cannot be lased by the designator operator; if, on the other hand, the target unmasks at a range less than the visibility limit, then the next test is made.

2.2.4 Designator Maximum Range. Using the above random sample from the acquisition (terrain unmask) range distribution, the next test compares the designator-to-target range with the maximum designator range. If the target is beyond the maximum designation range, then the potential COPPERHEAD fire mission is aborted; otherwise, the tests continue.

At night, the maximum designator range is reduced to the range to which the thermal imagery night sight is usable for aiming the designator.

2.2.5 Target Not Detected. The next test is performed to determine whether the target was in view long enough to be acquired (the entire detection, recognition identification process is lumped into one test and called detection).

After a range from designator-operator to target terrain unmask is obtained as above, a line-of-sight segment length distribution is sampled to obtain the length of the section of the target's path extending from the point of initial terrain unmask to the first point along the path that is obscured from the D.O.'s view. From this line-of-sight segment length together with the target velocity and length of the target unit column, the duration of line-of-sight is calculated. This duration is the time from initial target unmask until the last vehicle in the target unit's column passes out of the D.O.'s view.

To determine whether the target is detected, a time is sampled from a distribution of detection times. This time represents the delay from the time of initial target unmask until the D.O. has detected, recognized, and identified the target and decided to call for COPPERHEAD fire. This is termed detection time.

If the detection time is greater than the duration of the line-of-sight on a particular replication, then the target is considered to have gone in and then out of view before the D.O. could detect it (or really before he could call for fire against it) and the mission aborts. If the detection time is less than the line-of-sight duration, then the D.O. detects and calls for fire on the target before it leaves his line-of-sight and the model continues to the next test.

2.2.6 Smoke. The next test is to determine whether there is sufficient smoke in the area of the target to prevent the D.O. from either seeing or lasing the target. The smoke model computes the fraction of targets that could be obscured from view by taking into account such factors as number and type of RED smoke rounds fired and weather conditions for the month and time of day considered in the current case.

As an alternative to specifying a volume of RED smoke rounds which is then used to calculate the fraction of targets obscured, one can simply input a fractional value which represents the fraction of targets obscured by smoke.

In either case, the fraction of targets (actually fraction of a specified front width) obscured by smoke is played as a "probability of abort due to smoke." For each replication that reaches the smoke test, a random number is drawn from a uniform distribution (from 0 to 1) and if it is less than the probability of abort due to smoke, then the mission is aborted at this step; if it is greater than the probability of abort due to smoke, then the next test is performed.

Note that while the D.O. has a limited target detection capability in smoke by using a thermal site on the designator, the laser beam itself could not penetrate the smoke in those cases where smoke was thick enough to require use of a thermal site. Consequently, COPE plays a "black-white" smoke situation: either smoke completely aborts the mission or has no effect. This is not the case for some of the other COPE variants (such as TAPE, SOPE) where there are three possibilities: (1) smoke is not present and, hence has no effect, (2) smoke is present and kills mission, (3) smoke is present but target can still be detected through thermal site and target is engaged but with reduced accuracy.

2.2.7 Dust. The next test is whether the dust from RED HE artillery fire is sufficient to prevent the D.O. from either seeing the target or lasing it. This is modeled in a very simplistic way in the current generation of COPE.

A probability that dust aborts the mission is estimated outside the COPE program and then input for a given COPE case (either explicitly or by default). Then for each replication that reaches the dust test, a random number is drawn from a uniform distribution (0 to 1) and compared to the probability that dust aborts the mission. If the random number is less than the probability that dust aborts the mission, then the mission is aborted; otherwise, the next test is made.

2.2.8 Designator Bail-Out (Precommo). The next test is to check whether the target is so close to the D.O. that in order to survive he would not engage the target. The assumption is that to avoid being killed by direct fire from potential COPPERHEAD targets, the D.O. would move to a new position whenever the enemy front line units closed to within a certain range; in the case of an isolated lead enemy unit, he might merely refrain from designating rather than move, but in either case he would not be calling for COPPERHEAD fire.

To perform this test, the model compares the D.O.-to-target range against a pre-set "bail-out" range. If the D.O.-to-target range is less than the bail-out range, the mission aborts; otherwise, the mission proceeds to the next test.

Note that this bail-out check is made prior to the D.O.'s call for COPPERHEAD fire. A second bail-out check (see section 2.2.10) is made later.

2.2.9 Commo-out (Tests 8 and 9). The next test determines whether the D.O. has a communication link with the F.D.C.

Communication delay time and probability of success (Test 8) can be played in any one of three ways depending on the input option selected:

(1) A single response time distribution is used together with a single probability of having a good communications link. In this case, a number is randomly sampled from the response time distribution to determine the total delay time from the D.O.'s call for Copperhead fire until the firing battery is ready to pull the lanyard to fire Copperhead. In addition, a second random number is drawn, this time from a uniform (0 to 1) distribution; if this random number is less than the probability of having a good communication link, then the mission proceeds to the Test 9; otherwise, the mission is aborted.

(2) The second way of playing communication delay time and probability of good communication link is called the "parameterized response time" option. This case is essentially the same as (1) above except that the delay time is set at a single fixed value that is the same for every replication (as opposed to sampling a distribution of delay times to obtain a new delay time value for each replication).

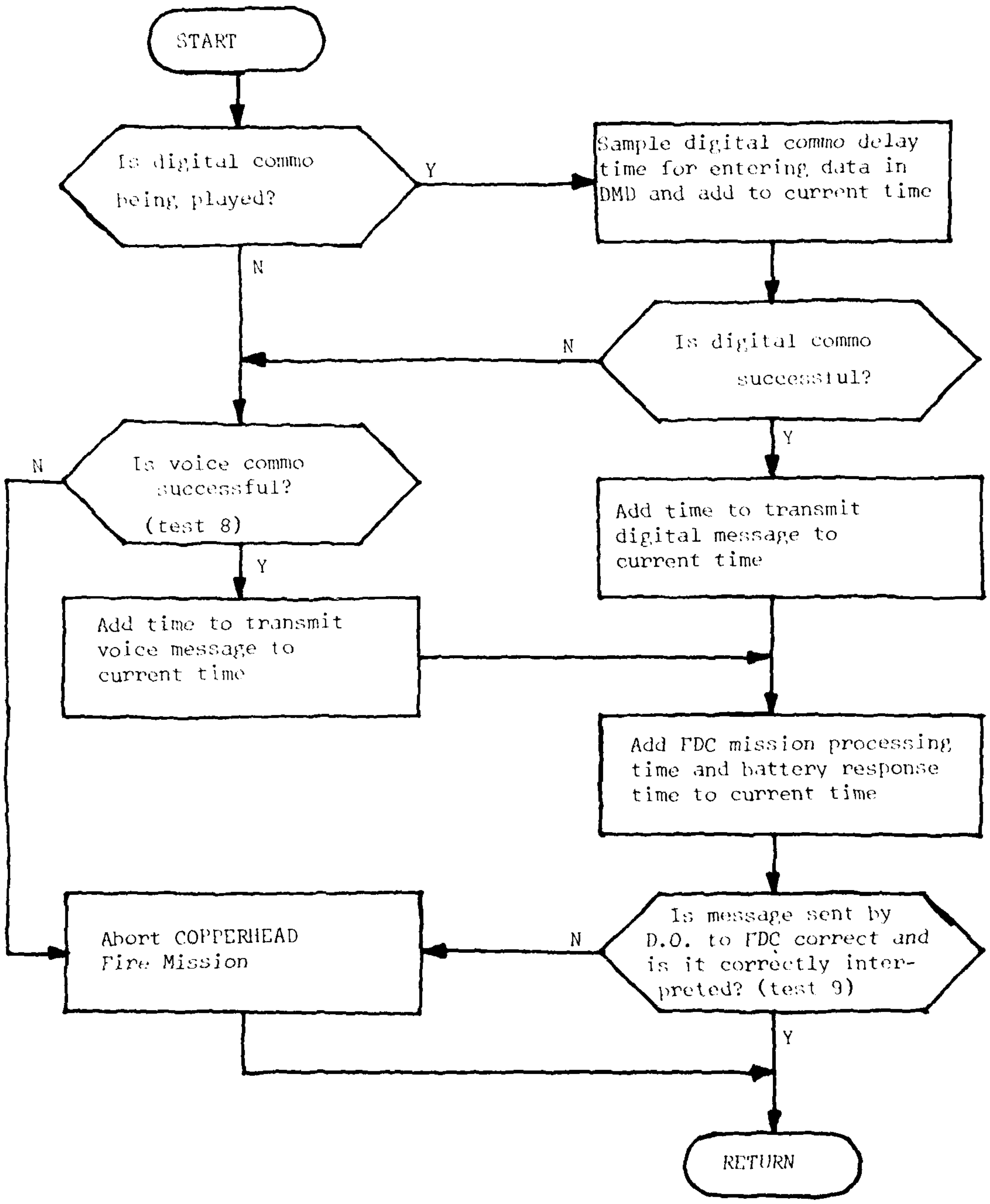



FIGURE 2-3 Third Communication Model

(3) The third way to model the communication delay time is to use a separate delay time for each part of the process from the D.O.'s calling in a fire mission to the battery's pulling the lanyard. Under this method, a delay time distribution is used to obtain digital communication times for D.O. to FDC messages. This delay time is a function of mission type and time of day (i.e., day or night).

The flowchart in Figure 2-3 gives the complete picture of the time delays when this third model is used.

No matter which of the three ways of modeling delay times is used, either communication fails and the mission aborts or communication is established and a delay time is added to the current time. Finally, if communication is established, another random number (uniform between 0 and 1) is drawn and compared to the probability of a successful (i.e., one containing the correct information) message (Test 9). If the random number is less than this probability, the mission proceeds to the next test; otherwise, the mission aborts because the message from the F.O. had bad data or was incorrectly interpreted.

2.2.10 Designator Bail-Out After Communication. After the communication from D.O. to FDC to firing battery is completed successfully, the D.O.-to-target distance is updated to account for target movement during the communication and processing delay. If the updated position brings the target within the "bailout" range of the D.O. (see 2.2.8) then the mission is aborted; otherwise, the mission continues with the next test.

2.2.11 Line-of-Sight Lost Prior to Firing. The next test is to determine whether the D.O. has lost LOS to the target. The current time (detection time plus communication and processing delay times) is compared to the duration of line-of-sight. If the current time is greater than the duration of line-of-sight, then the target has left the D.O.'s field of view prior to the firing of the first COPPERHEAD and the mission is aborted; otherwise, the mission is fired, the time of flight is added to the current time, and the next test is made.

2.2.12 Line-of-Sight Lost During Mission. The tests prior to this affected the entire mission; the sequence of tests after this point is performed separately for each round in the fire mission.

The current time (which is the time the first (next) COPPERHEAD round is to arrive) is compared to the duration of line-of-sight. If the current time is greater than the duration of line-of-sight, then the current round is aborted (ineffective); otherwise, the mission proceeds to the next test.

2.2.13 Designator Not Warned In Time. At the time the first COPPERHEAD round is fired for a given mission, a message is sent to the D.O. by the FDC telling him to begin lasing the target. To determine whether the D.O. receives this message, a random number is drawn from a uniform (0 to 1) distribution and compared to the probability of D.O. warned to lase. If the random number is less than the probability, then the message gets through, the D.O. lases, and the next test is performed; otherwise, the message does not get through, the D.O. fails to lase, and the first round is wasted (aborted).

Even if the message warning the D.O. to lase does not get through for the first COPPERHEAD round, the model assumes that the D.O. would see the first round impact and hence know to begin lasing for the second and subsequent rounds; therefore, rounds after the first round are never lost because of D.O.'s failure to lase.

2.2.14 Designator in Direct Fire. The next test determines whether the D.O. is unable to lase due to incoming direct fire. It is assumed that the target being lased for the COPPERHEAD fire mission has a laser alarm as well as capability that allows it to locate the D.O.'s position (when his laser is on) with sufficient accuracy to fire.

A random number is drawn from a uniform (0 to 1) distribution and compared to probability of kill and probability of suppression values (which are a function of D.O.-to-target range). If the random number is less than the probability of kill, then the D.O. is considered killed and the current round and all subsequent rounds for this replication are aborted. If the random number is greater than the probability of kill but less than the sum of the probability of kill plus the probability of suppression (obscuration due to direct fire), then only the current round is aborted as a result of this test (the test will be performed separately on subsequent rounds of this replication).

If the random number is greater than the probability of kill or suppression, then the mission continues to the next test with no effect from the direct fire.

2.2.15 Round Reliability Failure. The round is next tested for inflight reliability. A random number is drawn from a uniform distribution (0 to 1) and compared to the round reliability. If the random number is less than the reliability, the round is reliable and the next test is made; otherwise, the round fails.

2.2.16 Target Obscured By Mini-Terrain. The D.O.-to-target range is computed allowing for target movement up to the current model time (actually, the time the first (or next) COPPERHEAD round is to arrive). A distribution of target exposures (due to mini-terrain) at that range is then sampled to determine whether the target is fully exposed, hull defilade (turret exposed), or completely obscured. If the target is completely obscured, then the round aborts; otherwise, the next test is made.

2.2.17 Round Did Not Engage Target. The next test determines whether the COPPERHEAD round receives sufficient reflected laser energy to initiate maneuver to a target and has the target in its maneuver footprint.

To perform this test, a random number is drawn from a uniform (0 to 1) distribution and then compared to a probability of seeker engagement value which is a function of designator-to-target range, visibility range limit, cloud ceiling altitude, delay time, and number of vehicles in column as well as reflectivity of target, seeker sensitivity, target heading, target velocity, nominal COPPERHEAD system response time, gun-target-D.O. angle, designator type, and deflection biases (offsets). The probability of seeker engagement is the probability that both the seeker receives sufficient reflected laser energy to initiate tracking and the target is within the maneuver footprint of the COPPERHEAD round. If the random number is less than the probability of seeker engagement, then the next test is performed; otherwise, the round fails because its seeker cannot engage the target (or can engage but cannot maneuver to the target).

2.2.18 Round Engaged But Did Not Hit. A probability that the round hits the target given that it engages (i.e., passes the previous test) is obtained as a function of range, exposure (fully exposed or hull defilade), and designator type. A table of these probabilities is obtained as output from the LDWSS program and is intended to account for the possibility that the COPPERHEAD seeker tracks laser over-spill or under-spill rather than the target itself.

Again a uniform distribution (0 to 1) is randomly sampled. If the resulting random number is less than the probability that the round hits, then the round hits the target and we proceed to the next test; otherwise, the round misses and aborts.

2.2.19 Round Hit But Did Not Kill. The final test is to determine whether the round killed the target given that it hit the target. To perform this test, a random number is drawn from a uniform distribution (0 to 1) and compared to a probability of kill given a hit value which is a function of target type and target exposure (fully exposed or hull defilade). If the random number is less than the probability of kill, then the target is killed; otherwise, the target is not killed and the round aborts.

2.2.20 Subsequent Rounds. When a round (not a mission) aborts, subsequent tests in the sequence are not made. Instead, the time is advanced to the next round's arrival time (if there are more rounds left to be considered) and the sequence of tests resumes at the "LOS lost during mission" (section 2.2.12) test for the next round.

If a round does not abort, but instead successfully kills a target, then the time is advanced to the next round's arrival time (if applicable) and the sequence of tests resumes at the "LOS lost during mission" (section 2.2.12) test for the next round.

When no more rounds remain to be fired for a given replication, the replication is ended.

2.2.21 End of Replication. When either a mission (not a round) aborts or all the rounds for a replication have been fired, the replication is considered ended. If more replications remain to be made, the next replication begins with the first test, "Preparatory fires" (section 2.2.2) and proceeds through the sequence of tests as described. If all replications for the current case are done, then the results are printed out and the subsequent case (if any) is begun.

2.2.22 Random Occurrence LOS Model vs Shooting Gallery LOS Model. The step by step description given above of the various tests that must be passed for a successful COPPERHEAD fire mission applies to the "Shooting Gallery" or single continuous LOS segment model. This model assumes that the entire COPPERHEAD mission must be completed during the time interval from the first target vehicle's breaking terrain mask until the last target vehicle leaves that same LOS segment.

The alternate LOS model is called the "Random Occurrence" LOS model and assumes that after the first target vehicle breaks terrain mask, the target vehicles will go in and out of mask at random. The sequence of tests for the random occurrence method is the same as for the shooting gallery except that tests 4 and 11 (sections 2.2.5 and 2.2.11) are omitted and test 12 is modified.

Test 12 (section 2.2.12) is modified so that the probability that at least one target vehicle is visible is given by $Q = 1 - (1 - P)^N$ where P is the probability of a single vehicle being visible if randomly placed along an approach path at the current D.O.-to-target range and N is the number of vehicles left unkilled in the target. A random number is sampled from a uniform distribution (0 to 1) and compared to Q. If the random number is greater than Q, the round is aborted; otherwise, the mission continues with test 13.

2.2.23 Note on Nominal Response Time. The probability of engagement numbers are a function of the time after the D.O.'s call for fire that the rounds arrive. The time is played by establishing a nominal response time $\bar{t}_r$ (an average value for the distribution of total mission communication and processing times which is input and to which is added the time of flight appropriate for the given gun-to-target range.) The model assumes that the D.O. estimates the target's location at $\bar{t}_r$ seconds in the future and calls for fire against that point. The actual COPPERHEAD mission response time is sampled and added to the time of flight to obtain the actual arrival time of the round ($t_a$). If $t_a$ is greater than $\bar{t}_r$, then the probability of engagement is lowered because of the larger aiming error and TLE.

This is explained in greater detail in the report on the PAM model. The report on modeling target location error (TLE) discusses the problem of estimating target location and the error in the estimate as a function of time (see bibliography).

The only point important to the user at this stage is that COPE uses two response times: a nominal response time $\bar{t}_r$ and an actual response time $t_a$. The nominal response time is used to estimate target's future location and hence to choose the point at which to aim. The actual response time is used to determine the probability of seeker engagement (PE). Naturally, if $t_a$ is very large, the probability of the target being at the point estimated (based on $\bar{t}_r$) is quite small.

Also, it was assumed that if the firing battery saw that $t_a$ was likely to be less than $\bar{t}_r$ on a particular occasion, it would hold up firing to cause $t_a = \bar{t}_r$ rather than drop early rounds far in front of the target. This assumes a good communication capability between D.O. and battery.

# CHAPTER 3

## 3. PERFORMANCE AND LIMITATIONS

### 3.1 Computer Related Performance and Limitations.

The COPE program (and its preprocessor programs) has been run primarily on the U.S. Army Ballistic Research Laboratory's Control Data Corporation CYBER 76 computer (CDC 7600). On that computer, a typical case with 10,000 replications and six rounds fired per successful COPPERHEAD mission requires 2.5 to 3.0 seconds per case. The computer memory requirement is about 24,000 words (57 Kg) of SCM and no LCM. The disk space required (which will vary with the size of the data base file) for TAPE 11 was about 155,000 words (456 Kg).

COPE has also been run on a CDC 6600 at Aberdeen Proving Ground and on a CDC 6400 and a CDC 6500 at FT. Leavenworth (using the remote terminal at FT. Sill). On these machines, the run time ranged from eight to sixteen times what it was on the CDC 7600. The memory requirements and disk space on the various CDC 6000 series computers were nearly the same as for the CDC 7600.

At the time of this writing, COPE has not been run on any non-CDC machines; however, chapter 13 of this manual considers some of the changes required to do so. Run time would, of course, vary from machine to machine (e.g., the author's experience with other FORTRAN programs suggests run time on the UNIVAC 1108 computer would be eight to ten times that on the CDC 7600).

Most current non-CDC computers use a word size of fewer than 60 bits (the CDC word size). This means that the program memory required to load the object program on most non-CDC machines will be somewhat larger in terms of number of words than for the CDC computers. Similarly, a slight increase in disk space may be required to put the data base file TAPE 11 on a smaller word machine.

### 3.2 Program Related Performance and Limitations.

The current version of COPE has been run several hundred times now with no errors detected other than those resulting from invalid input, insufficient run time requested, or insufficient print limit set.

The ability of the program to run multiple cases with relatively small streams of input directives has enabled the users to run hundreds of cases with relatively short set up time.

The main limitation encountered in running the program is the requirement that data selected to fill the eight data blocks be on the random access mass storage data base file created by the preprocessor

programs. Once these preprocessors are run, any case using data already on the data base file can be easily run. This subject is discussed in detail in chapter 5.

## 3.3 Model Related Performance and Limitations.

One of the main limitations of the COPE model is that much tactical detail is represented by distributions. For this reason, great care must be taken to interpret input distributions when reporting model results. The interpretation of results depends heavily on the conservatism and objectivity of the analyst.

The only attempt (and indeed the only opportunity) to verify the COPE model was by running it with case descriptors matching as closely as possible conditions present in the COPPERHEAD OT II. When this was done, the results predicted by COPE were compared to those obtained in OT II and found to be in quite close agreement. More precisely, the hypothesis that the probability of hit given a shot as obtained from COPE is equal to the true population probability of hit given a shot could not be rejected even with a significance level as high as 50 percent based on the sample of firings in OT II.

This good agreement of the model to test results should not lead to uncritical acceptance of all COPE results. For one thing, tests 1, 5, 6, and 14 were nulled out (that is, always passed). Also, since the OT II rounds were not live, the probability of kill values (test 19) in the model were set to one (so that kills and hits were considered equivalent).

There are some limitations to the COPE model that need to be explicitly pointed out:

3.3.1 Geometry of COPPERHEAD Fire Mission. The geometry of the gun, D.O., and target positions is only approximated. For any given simulated COPPERHEAD mission, the gun-to-target range, designator-to-target point-of-unmask range, target heading, and nominal gun-target-designator angle are fixed while the target itself moves (i.e., its position is a function of time).

Figure 3-1 represents the geometry of a COPPERHEAD fire mission.

U is the point of target unmask.

R is the point at which round is to arrive on target (provided nominal response time is met).

D is the designator operator location.

G is the gun location.

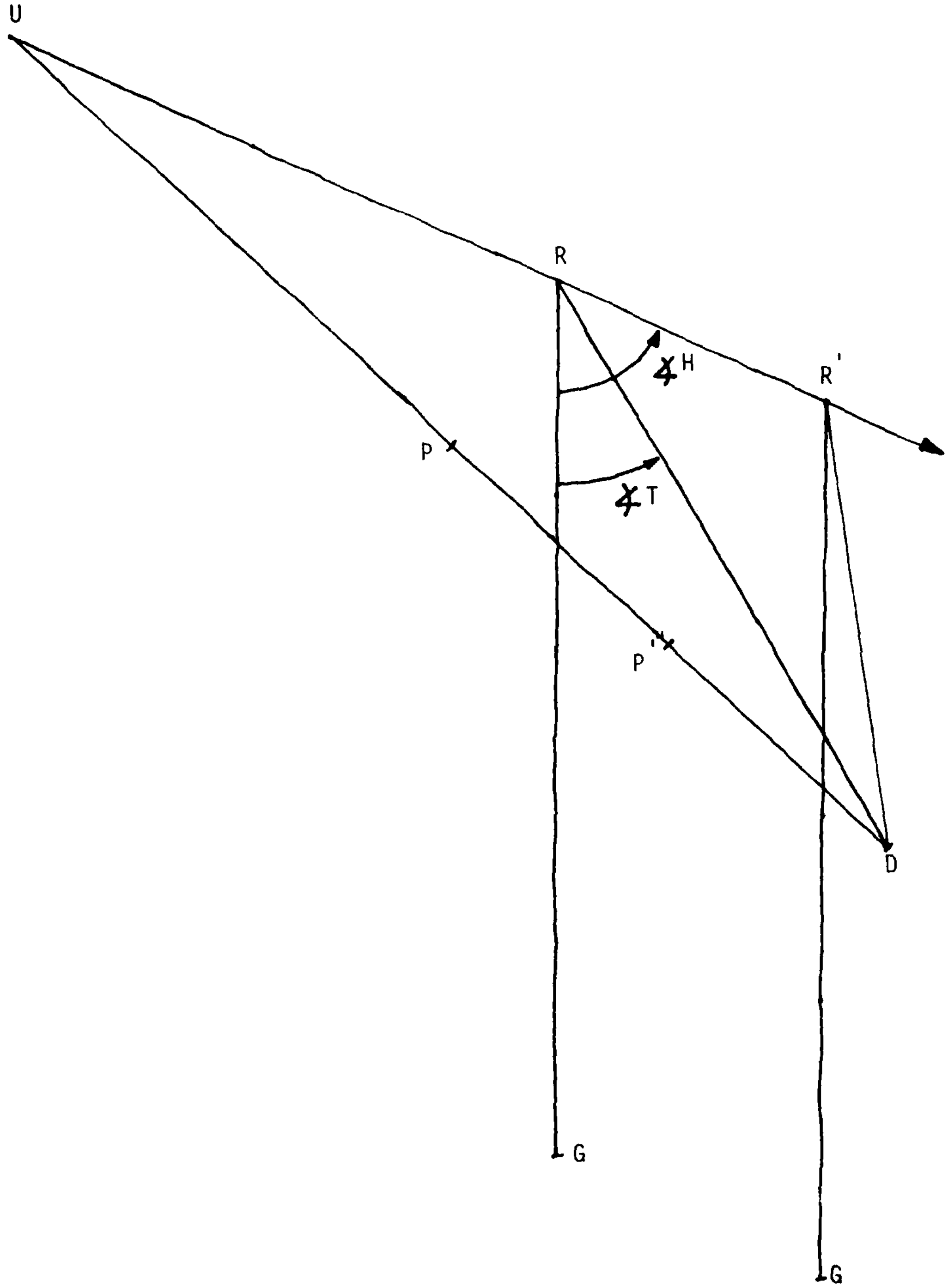



FIGURE 3-1 Geometry of COPPERHEAD Fire Mission

∡H is the target heading (angle between gun-target line and target path at time round arrives on target).

∡T is the angle between the gun-to-target line (GR) and the designator-to-target line ($\overline{DR}$).

UR is the target's path (target travels from U to R).

By the COPE case conditions chosen, ∡T, ∡H, the length of $\overline{GR}$, and the length of $\overline{UD}$ are constant (the first three are constant for a given case; the last, length of $\overline{UD}$, is constant for a given replication, but is sampled anew for each replication). Hence, if the response time is such that the target is at R' on one replication rather than at R (as on, say, a previous replication) when the round arrives, one must move G to G', and D to D' to preserve the fixed values of ∡T, ∡H, length of $\overline{GR}$ (now G'R').

The requirement that G move to G' is, of course, unrealistic since one does not move the battery to a new site just because nominal response time is not met.

A second compromise in the geometric representation of the COPPERHEAD mission is in calculating the designator-to-target range. Rather than compute the designator-to-target range $\overline{DR}$ (or $\overline{DR'}$ in the second case) in the correct manner, the COPE model approximates the designator-to-target range by the distance $\overline{PD}$. P is a point chosen on $\overline{UD}$ such that $\overline{UP}$ = $\overline{UR}$ (or P' such that $\overline{UP'}$ = $\overline{UR'}$ in the second case). This approximation is equivalent to the assumption that the target moves straight toward the designator.

One can see from the line segment lengths in Figure 3-1 that $\overline{PD}$ does not differ greatly from $\overline{RD}$ (in this case) whereas $\overline{P'D}$ does differ considerably from $\overline{R'D}$ in length. For typical range values and velocities used so far in COPE, these errors are considered minor. If one is to use greater target velocities or shorter acquisition ranges, then it would be worthwhile to improve this feature of COPE. (That is, incorporate the trigonometry required to correctly calculate $\overline{DR}$ (or $\overline{DR'}$) and use that value instead of $\overline{PD}$ (or $\overline{DP'}$).

Note: The designator operator-to-target distance is used not only at round arrival to calculate probability of engagement and probability of hit; but also throughout the simulation it is updated and used in checking bail-outs, direct fire suppression, line-of-sight probability for random occurrence, etc.

3.3.2 Target Modeling. The COPE model makes simplifying assumptions about the target. In particular, it assumes that the target is a column of vehicles moving at constant velocity with uniform spacing between successive vehicles. When a vehicle is killed, the length of the column is not reduced in length nor is any attempt made to keep a record of the position that the killed vehicle occupied in the column.

These limitations were not deemed serious in the COPE runs for the COPPERHEAD COEA; however, for the SADARM variant of COPE (called SOPE) the model was modified to keep track of individual vehicle status and to handle more general target layouts. (These modifications were essential to SADARM where there is no designator operator to pick out live targets instead of dead ones, and where the volley attacks an area instead of a point as with a single COPPERHEAD round).

3.3.3 Direct Fire Effects Independent. The model of direct fire effects of the target against the D.O. (see section 2.2.14) is over-simplified. The model has a single probability of kill (or obscuration) by direct fire for each target-to-D.O. range and this probability (corrected for then current range) is then applied during each COPPERHEAD round's firing to determine whether the D.O. is killed (or obscured) and thus prevented from lasing.

The shortcomings of this model are (1) subsequent rounds fired by the target at the D.O. should be more accurate both due to adjustment (assuming both that the D.O. survived previous rounds fired by the target and that the target survived previous COPPERHEAD's) by the target after seeing its first round impact and due to the D.O.'s giving away his position more times by the additional lasings, and (2) adjacent vehicles in the target unit, after seeing the lasings for the first few rounds, might join in firing at the D.O. on the latter rounds. Both effects (1) and (2) are omitted in COPE's model of direct fire against the D.O..

3.3.4 Only Initial Acquisitions (Unmaskings) Played. The COPE model only plays the first unmasking of each target unit. A more realistic assessment of COPPERHEAD's contribution would be obtained if the target unit were acquired (or, at least, unmasked to provide a potential acquisition) several times as it closes toward the BLUE position. This would require keeping track of the damage done to the RED target unit as the result of previous unmaskings ( and the resulting COPPERHEAD fire missions) so as not to take credit for killing vehicles many times over. It would be a fairly straight forward task to modify the COPE program to allow for multiple acquisitions (and firing) of each target unit. The reason this was not done in the original COPE model is simply that all of the acquisition data (range to unmask and line-of-sight segment lengths) available to the model's creators were for first acquisition only.

3.3.5 Other Limitations. Other limitations to COPE are chiefly due to either inadequate data or to the desire to keep the model and program simple. These limitations include:

(a) Availability of artillery to fire the COPPERHEAD mission is not considered.

(b) The impact of the attrition of a D.O. or an F.O. (forward observer) on the ability of the direct support artillery to perform its mission is not considered. That is, how much of the lost D.O. capability could be

picked up by neighboring D.O.'s, other Fire Support Team (FIST) members, or platoon F.O.'s?

(c) The permanent loss of communications is not played. This could be partially simulated by reducing the probability of a successful transmission, although that value was intended to represent only degradation due to high communication traffic load and electronic warfare.

CHAPTER 4

# 4. INPUT DIRECTIVES AND "DATA" STATEMENTS

## 4.1 Input.

Considerable effort has gone into making the COPE input versatile enough to allow for the running of nearly any combination of input parameters while at the same time requiring a minimum of input changes to switch from one case to another. The stacking of multiple cases for a single run can be easily accomplished with relatively few input cards.

The existence of "default" (baseline case) values built into the program allows the program to be run with only the input cards (or card images) required to change those variables which it is desired to set at other than the default values. If further cases are desired one can continue with additional input changes applied to either the previous case or the baseline case.

The program also allows the insertion of up to three types of user comments which may appear in the output.

Finally, in the event that one wishes to use data not in the "data base" file, a "temporary" option using traditional formatted data is available.

## 4.2 Data Organization.

For each case that is run, there are numerous parameters whose values are set to provide the mathematical description of that case. These parameters are grouped into eight major blocks, six condition keys, and twenty odd individually selected numerically valued variables.

For the most part, a single input card is used to select each block, key, or variable value. There are some exceptions and some dependencies between choices which are explained below (section 4.8).

Finally, there are numerous data items whose values are set in the Block Data (BDATA1) subprogram because they are relatively fixed from case to case. Obviously, if one chose to vary these items frequently, they could be made into input rather than set via data statements. These items are defined below (section 4.7)

## 4.3 Keyword Input: General Description.

Each input to select a case description (except when the "temporary" option is used) as well as various inputs to control program options is of the form:

keyword, option word 1, option word 2, , option word n
where $0 \leq n \leq 9$. Each such input must be contained on one card (or card image); no continuation to another card is allowed.

The keyword is an alphanumeric character string indicating which case descriptor is to be set by the card.

The option words are either alphanumeric character strings that indicates which option is chosen from a pre-defined set or numerical values that are to be used for certain variables.

Option words are order dependent (i.e., keyword, x, y is not equivalent to keyword, y, x). If an option word is omitted, a default value is set by the program. If a keyword takes two or more option-words and an option word other than the last one is omitted, then the proper number of separators (commas, for example) must be used to position the later optionwords (e.g.,

keyword, default value for option word 1, option word 2

is equivalent to

keyword, , option word 2

but not equivalent to

keyword, option word 2 ).

In most cases, blanks may be freely inserted within keywords or option words to improve readability. (The only exception is covered in the note of section 4.5.3.).

## 4.4 Keyword Input: Specifics.

Now each keyword is shown with the option words allowed. Note that new option words may be introduced by running the PRLPMS pre-processor program to modify the data base file (section 5.3.8 and chapter 14).

### 4.4.1 FIRST CASE.

(a) general form: FIRST CASE

(b) This card is used only as the first non-comment card in an input deck. If it is omitted, the program will generate a message upon encountering the first non-comment card in the input and then proceed as though the FIRST CASE card were present.

4.4.2 NEXT CASE.

(a) general form: NEXT CASE, option word 1

(b) Currently allowed character strings for option word 1 are:

(1) RESET DEFAULTS or RD

(2) DON'T RESET or DR.

(c) This card is the first non-comment card of each case other than the first case.

If the option word is RESET DEFAULTS, RD, or omitted, then the values for all case description parameters are reset to their default values and will differ from the baseline values only as determined by the remaining input cards for the current case.

If the option word is DON'T RESET or DR, then the values of the case description parameters remain at their values for the previous case except for those values changed by the remaining input cards for the current case.

If the NEXT CASE card is omitted, the program will generate a message and proceed as though a NEXT CASE, DR card had been encountered.

Note: The PLOT and CONTROL cards (sections 4.4.5 and 4.4.6) are not controlled by the option on the NEXT CASE card.

4.4.3 END.

(a) general form: END

(b) This card belongs at the end of the input for each case other than the final case.

If it is omitted, an error will usually result because the program will continue reading cards that are really part of the next case's input until it either hits an END card, ENDF card, or the end of the input file. This means that the case will run using its own input together with input from subsequent case (or cases). The user can detect this problem from the fact that the case descriptors will probably be incorrect (i.e., other than the intended values) in the case heading print-out, there may be unexpected messages stating that certain input cards have superseded others, and finally, the number of cases in the run will be wrong.

4.4.4 ENDF.

(a) general form: ENDF or END OF FINAL CASE.

(b) This card should be the final card of the final case.

If it is omitted, the program will generate a message on reaching the end of input and then proceed as though an ENDF had been encountered.

If an END card is used instead, there will be no change in the result but the program will waste a few nano-seconds before realizing that there is no more input.

If an ENDF is used prior to the end of the last case, any cards following the ENDF card will be ignored.

4.4.5 PLOT.

(a) general form: PLOT, option word 1, option word 2

(b) Allowed character strings for option word 1 are:

(1) PRINTER

(2) CALCOMP.

Allowed character strings for option word 2 are:

(1) OFF

(2) ON.

(c) This keyword was designed to handle graphic output (bar charts) , but has not been implemented in the current version of the program.

The cards PLOT, PRINTER, ON and PLOT, CALCOMP, ON would cause results to be displayed on bar graphs produced by the line printer and Cal-Comp plotter respectively. At the present time, however, these cards will only cause the program to generate a message saying that no plot was produced.

The cards PLOT, PRINTER, OFF and PLOT, CALCOMP, OFF turn off the production of the bar graphs produced by the line printer and Cal-Comp plotter respectively.

The default for option word 1 is PRINTER; the default for option-word 2 is OFF. So no bar graphs (or, at the present time, "no plot" messages) are produced unless positive action is taken by the user.

Note, that if and when this feature is implemented, the two types of graphs may be selected independently of each other.

A PLOT option once selected remains in effect until a further PLOT card is encountered changing the option (that is, ON to OFF or OFF to ON). It is not affected by a NEXT CASE, RD card.

4.4.6 CONTROL.

(a) general form: CONTROL, option word 1, option word 2

(b) Allowable choices for option word 1 are:

(1) ECHO

Allowable choices for option word 2 are:

(1) SHORT

(2) LONG.

(c) This keyword is designed to control miscellaneous program features. At the present time, the only choice (and the default) for option word 1 is ECHO. If option word 2 is SHORT or omitted, then the short form of the input echoing is used; if option word 2 is LONG, the long form of the input echoing is used (see section 7.6 for input echoing).

The default (which occurs if the card is omitted) is the short form.

A CONTROL option once selected remains in effect until a further CONTROL card is encountered changing the option (that is, LONG to SHORT or SHORT to LONG). It is not affected by a NEXT CASE, RD card.

4.4.7 LINE OF SIGHT MODE.

(a) general form: LINE OF SIGHT MODE, option word 1 or LOS MODE, option word 1

(b) Allowable choices for option word 1:

(1) SHOOTING GALLERY or SG

(2) RANDOM OCCURRENCE or RO.

(c) The choice of SHOOTING GALLERY or SG for option word 1 causes the current case to be run using the "shooting gallery" line of sight model. If option word 1 is RANDOM OCCURRENCE or RO, then the "random occurrence" line of sight model is used.

The default is the "shooting gallery" model.

4.4.8 MISSION CODE

(a) general form: MISSION CODE, option word 1 or MSNCODE, option word 1

(b) Allowable character strings for option word 1 are:

(1) PRIORITY PREPLANNED TARGET or PPPT

(2) PREPLANNED TARGET or PPT

(3) TARGET OF OPPORTUNITY or TOO.

(c) If option word 1 is PRIORITY PREPLANNED TARGET, PPPT, or omitted, then the mission is considered priority preplanned; if option word 2 is PREPLANNED TARGET or PPT, then the mission is considered preplanned; and, if option word 2 is TARGET OF OPPORTUNITY or TOO, then the mission is considered a target of opportunity.

The mission type affects the response time parameters used (see section 4.4.15) and the modeling of range from unmask to centroid (RUMC, section 4.4.18). See section 4.8 for side effects of mission code choice.

4.4.9 NUMBER OF DIFFERENT TARGET POSTURES.

(a) general form: NUMBER OF DIFFERENT TARGET POSTURES, option word 1 or NUMDIFTGTP, option word 1

(b) Allowable character strings for option word 1 are:

(1) THREE or 3

(2) TWO or 2.

(c) If option word 1 is THREE, 3, or omitted, the target exposure (mini-terrain) model plays three exposures or postures (fully exposed, hull defilade, and completely obscured).

If option word 1 is TWO or 2, the target exposure model plays two exposures (fully exposed and hull defilade) with the completely obscured divided between the other two postures in proportion to their relative fractions.

The default option word 1 is THREE.

4.4.10 TARGET TYPE.

(a) general form: TARGET TYPE, option word 1

(b) Allowable character strings for option word 1 are:

(1) 1

(2) 2

(3) 3

(4) 4

(5) 5

(6) 6

(7) 7

(c) If option word 1 is i, then the vulnerability values (probability of kill given a hit as a function of target exposure) for target type i are used.

The default target type is 1.

4.4.11 DESIGNATOR TYPE.

(a) general form: DESIGNATOR TYPE, option word 1 or DESIGTYPE, option word 1

(b) Allowable choices for option word 1 are:

(1) GLLD or G

(2) MULE or M

(3) LTD or L

(c) If option word 1 is GLLD, G, or omitted, the designator type is the GLLD; if option word 1 is MULE or M, the designator type is the MULE; and, if option word 1 is LTD or L, the designator type is the LTD.

The default designator type is GLLD.

See section 4.8 for side effects of Designator Type Choice.

4.4.12 DO POSITION.

(a) general form: DO POSITION, option word 1

(b) Allowable choices for option word 1 are:

(1) VANTAGE POINT or VP

(2) MANEUVER UNIT or MU.

(c) If option word 1 is VANTAGE POINT, VP, or omitted, then the target posture table gives break down into target exposure as seen from a vantage point.

If option word 1 is MANEUVER UNIT or MU, then a target posture table is used which gives the breakdown into target exposures as seen from a position down with the maneuver units.

The default for DO POSITION is VP.

4.4.13 WEATHER.

(a) general form: WEATHER, option word 1, option word 2 or W, option word 1, option word 2

(b) Allowable choices for option word 1 are:

(1) JUNE or J

(2) DECEMBER or D

(3) MARCH or M

(4) SEPTEMBER or S

Allowable choices for option word 2 are:

(1) 1400 or 14

(2) 0600 or 06

(3) 2200 or 22.

(c) The weather data used in COPE is grouped into sets of data; each set corresponding to a certain month and time-of-day combination.

The choice for option word 1 determines the month whose weather data are to be used for the current case.

The choice for option word 2 determines the time-of-day whose weather data are to be used for the current case.

The defaults are JUNE or J for option word 1 and 1400 or 14 for option word 2.

See section 4.8 for side effects of weather choice.

4.4.14 ACQUISITION RANGE DISTRIBUTION.

(a) general form: ACQUISITION RANGE DISTRIBUTION, option word 1 or ACQRNGDIST, option word 1

(b) Allowable choices for optionword 1 are:

(1) AVERAGE or A

(2) CLOSE or C

(3) OPEN or O.

(c) If option word 1 is AVERAGE, A, or omitted, then the acquisition range and line-of-sight segment length distributions used are those for TETAM "average" terrain. If option word 1 is CLOSE or C, then distributions for TETAM "close" terrain are used. If option word 1 is OPEN or O, then distributions for TETAM "open" terrain are used.

The default is to use TETAM "average" terrain.

See section 4.8 for side effects of acquisition data choice.

4.4.15 RESPONSE TIME.

(a) general form: RESPONSE TIME, option word 1, option word 2, option word 3, or RESPTIME, option word 1, option word 2, or option word 3

(b) Allowable choices for option word 1 are:

(1) FTSILL or FS

(2) STAUCH or S

(3) SCUNGIO

(4) PARAM

Allowable choices for option word 2 are:

(1) DIGITAL or D

(2) VOICE or V

(3) a positive numerical value (which represents response time in seconds).

Allowable choices for option word 3 are:

(1) DAY or D

(2) NIGHT or N.

(c) If option word 1 is FT SILL, FS, or omitted, then the response time distribution supplied by USAFAS at FT Sill is used. This plays the first way of modeling delay time as described in Section 2.2.9.

If option word 1 is STAUCH or S, the response time numbers are obtained from the BCS DT II for some processes and TACFIRE tests for others. These data were supplied by Ed Stauch of AMSAA and use the third way of modeling delay time as described in Section 2.2.9.

If option word 1 is SCUNGIO, then the response time distribution used is a hypothetical distribution created by Richard Scungio of AMSAA to test the sensitivity of COPPERHEAD performance to longer response time. In this case the third modeling of response time (see Section 2.2.9) is used.

If option word 1 is PARAM, then the second way of modeling response time delay (see Section 2.2.9) is employed (so-called "parameterized" response time). In this case, the response time is a constant equal to the positive numerical value used as option word 2.

If option word 2 is DIGITAL, D, or omitted, then digital communication is used (with voice back-up when using the third way of modeling in Section 2.2.9).

If option word 2 is VOICE or V, then voice communication is used (this will have an affect only when used with the third way of modeling in Section 2.2.9)

If option word 2 is a positive numerical value, then this value is used as the "parameterized" response time. This can only be used with the PARAM choice for option word 1. A zero or blank option word 2 value used with PARAM as option word 1 will cause the program to stop (see section 7.5.18).

(Note: If one really wishes to run a zero parameterized response time value, a value of .001 could be used. The difference between these results and those obtained with a true zero response time would be far below the noise in the model).

If option word 3 is DAY, D, or omitted, day time response times are used. If option word 3 is NIGHT or N then nighttime response times are used. In addition, the choice of day or night influences the target velocity and maximum designator range (see Sections 4.4.21 and 4.4.22).

The defaults are: for option word 1, FTSILL; for option word 2, DIGITAL; for option word 3, DAY.

See section 4.8 for side effects of response time data choice.

4.4.16 DIRECT FIRE SUPPRESSION.

(a) general form: DIRECT FIRE SUPPRESSION, option word 1
or DFIRESUPPR, option word 1

(b) Allowable choices for option word 1 are:

(1) HIGH or H

(2) NONE or N

(c) If option word 1 is HIGH, H or omitted, then the direct fire suppression (obscuration from direct fire) and kill table used is for a high level of D.O. suppression and kill (corresponding to tank fire at the D.O.)

If option word 2 is NONE or N, then there is no effect of direct fire on the D.O..

The default is HIGH.

4.4.17 INVARIANT.

(a) general form: INVARIANT, option word 1

(b) Allowable choices for option word 1:

none

(c) The only currently allowed choice for invariant data is the default. Provision for other choices is built into the program for possible future use and may be activated by appropriate preprocessor run(s).

4.4.18 RUMC.

(a) general form: RUMC, option word 1

(b) Allowable choices for option word 1 are:

(1) NO or N

(2) a numerical value.

(c) The RUMC (which stands for range from unmask to centroid) value is the distance from the point at which the target breaks terrain mask to the centroid of the COPPERHEAD footprint.

If option word 1 is NO, N or omitted, then RUMC is assumed to vary with each replication in such a way that the centroid of the footprint is set at the predicted target (first vehicle) location for first round arrival. This predicted location is based on the nominal (estimated) response time (including time-of-flight) and the target velocity.

If option word 1 is a numerical value, then RUMC remains at that value for each replication.

The RUMC keyword has an effect only when the MISSION CODE is PPPT or PPT (see Section 4.4.8). If the MISSION CODE is TOO, then the program behaves as though option word 1 were NO regardless of what may be input on a RUMC card.

The default for RUMC is NO.

### 4.4.19 SMOKE.

(a) general form: SMOKE, option word 1, option word 2, option word 3

(b) Allowable choices for option word 1:

(1) SMOKE ROUNDS FIRED or SMKRNDS

(2) PROBABILITY SMOKE KILLS MISSION or PRBSMOKE.

Allowable choices for option word 2:

(1) a numerical value

Allowable choices for option word 3:

(1) a numerical value.

(c) If option word 1 is SMOKE ROUNDS FIRED, SMKRNDS, or omitted, then option words 2 and 3 respectively give the numbers of type 1 smoke rounds fired and type 2 smoke rounds fired.

If option word 1 is PROBABILITY SMOKE KILLS MISSION or PRBSMOKE, then option word 2 gives the probability that smoke kills the potential COPPERHEAD fire mission.

The default for option word 1 is SMKRNDS. The default for option word 2 is 484 when option word 1 is SMOKE ROUNDS FIRED or SMKRNDS, but it is .40 when option word 1 is PROBABILITY SMOKE KILLS MISSION or PRBSMOKE.

The default for option word 3 is 846 when option word 1 is SMOKE ROUNDS FIRED or SMKRNDS; it is unused otherwise.

Note: When the first alternative for SMOKE option word 1 is chosen, the probability that smoke aborts the potential COPPERHEAD mission is calculated as a function of the number of smoke rounds of each type fired and the weather conditions.

When the second alternative is used, the value input for option word 2 (a probability between 0 and 1) is used as the probability that smoke aborts the potential COPPERHEAD mission. (See section 2.2.6)

4.4.20 DO ARTILLERY PK.

(a) general form: DO ARTILLERY PK, option word 1 or DOARPK, option word 1

(b) Option word 1 must be either a numerical value between 0 and 1 or omitted.

(c) The numerical value input for option word 1 is used as the probability that the D.O. is killed by RED preparatory artillery fire.

The default value is .01.

4.4.21 TARGET VELOCITY.

(a) general form: TARGET VELOCITY, option word 1 or TGTVEL, option word 1

(b) Option word 1 is a positive number which gives the target velocity in meters per second. (The program is currently set up to accept only 2, 3, 5, 8, or 9 as allowed velocity).

(c) The value of option word 1 is used as the target velocity.

The default target velocity is obtained from a table (part of the acquisition range data block) giving target velocities as a function of day or night, weather, and terrain type.

Note: For side effects of velocity choice, see section 4.8.

4.4.22 MAXIMUM DESIGNATOR RANGE.

(a) general form: MAXIMUM DESIGNATOR RANGE, option word 1 or MAXDESRNG, option word 1

(b) Option word 1 is a positive number giving the maximum designator range in meters. This is the designator range mentioned in section 2.2.4.

The default maximum designator range is obtained from a table giving ranges as a function of designator type and day or night.

Note: For side effects of designator range value, see section 4.8.

4.4.23 BAIL OUT RANGE.

(a) general form: BAIL OUT RANGE, option word 1 or BAILOUTRNG, option word 1

(b) Option word 1 is a positive number giving the bail-out range in meters ( see section 2.2.8 and 2.2.10). The default bail-out range is 1000m.

4.4.24 TIME OF FLIGHT.

(a) general form: TIME OF FLIGHT, option word 1 or TOF, option word 1

(b) Option word 1 is a positive number giving the time of flight of the COPPERHEAD round in seconds.

The default value for time of flight is obtained from a table giving times of flight as a function of gun-to-target range.

Note: For side effects of time of flight value, see section 4.8.

4.4.25 ANGLE T.

(a) general form: ANGLE T, option word 1

(b) Option word 1 is a number giving the angle (in degrees) between the gun-to-target line and the designator-to-target line. The default value is 25 degrees.

Note: For side effects of angle T value, see section 4.8.

4.4.26 DUST.

(a) general form: DUST, option word 1

(b) Option word 1 is a number from 0 to 1 whose value is the probability that dust kills the potential COPPERHEAD mission.

The default value is .37.

4.4.27 PROBABILITY OF SUCCESSFUL VOICE TRANSMISSION.

(a) general form: PROBABILITY OF SUCCESSFUL VOICE TRANSMISSION, option word 1 or PRBVOCTRAN, option word 1

(b) Option word 1 is a number from 0 to 1 whose value is the probability that voice communication is established between the D.O. and the battery FDC.

The default value is .975.

4.4.28 PROBABILITY OF SUCCESSFUL DIGITAL TRANSMISSION.

(a) general form: PROBABILITY OF SUCCESSFUL DIGITAL TRANSMISSION, option word 1 or PRBDGTTRAN, option word 1

(b) Option word 1 is a number from 0 to 1 whose value is the probability that digital communication is established between the D.O. and the battery FDC.

The default value is .915.

4.4.29 ROUND IN FLIGHT RELIABILITY.

(a) general form: ROUND IN FLIGHT RELIABILITY, option word 1 or RNDFLTREL, option word 1

(b) Option word 1 is a number between 0 and 1 whose value is the probability that the round is reliable (functions properly).

The default value is .96.

4.4.30 TIME BETWEEN ROUNDS.

(a) general form: TIME BETWEEN ROUNDS, option word 1 or TBR, option word 1

(b) Option word 1 is a number whose value is the time (in seconds) between the firing of successive COPPERHEAD rounds.

The default value is 20 seconds.

4.4.31 NUMBER OF REPLICATIONS.

(a) general form: NUMBER OF REPLICATIONS, option word 1 or NUMREP, option word 1

(b) Option word 1 is the number of replications (i.e., Monte Carlo sample size of potential COPPERHEAD fire missions) to be used for the current case.

4.4.32 NUMBER OF ROUNDS TO BE FIRED.

(a) general form: NUMBER OF ROUNDS TO BE FIRED, option word 1 or NUMRNDS, option word 1

(b) Option word 1 is the number of rounds to be fired on each potential COPPERHEAD fire mission of this case that reaches the point where firing occurs (i.e., a "snot" event). Since each round is fired and completes its flight (successfully or not) before the next round arrives, one may choose to think of the number of rounds to be fired as the number of one round volleys to be fired.

The default value is 6 rounds.

4.4.33 NUMBER OF VEHICLES PER TARGET.

(a) general form: NUMBER OF VEHICLES PER TARGET, option word 1 or NUMVEHTGT, option word 1

(b) Option word 1 is the number of vehicles in the target column.

The default value is 10 vehicles.

4.4.34 DISTANCE BETWEEN VEHICLES.

(a) general form: DISTANCE BETWEEN VEHICLES, option word 1 or DISTBVEH, option word 1

(b) Option word 1 is the mean distance (in meters) between adjacent vehicles in the target column.

The default value is 61 meters.

4.4.35 GUN TARGET RANGE.

(a) general form: GUN TARGET RANGE, option word 1 or GTRNG, option word 1

(b) Option word 1 is the gun-to-target range in kilometers.

The default value is 8 kilometers.

Note: For the side effects of gun-to-target range, see section 4.8.

4.4.36 REFLECTIVITY.

(a) general form: REFLECTIVITY, option word 1 or REFLECT, option word 1

(b) Option word 1 is a number between 0 and 1 whose value is the reflectivity of the target to laser energy (i.e., it is the fraction of the laser energy at the designator's frequency that is reflected by the target).

The default value is .10.

Note: The side effects of reflectivity are discussed in section 4.8.

4.4.37 DEFLECTION BIAS.

(a) general form: DEFLECTION BIAS, option word 1 or DEFB, option word 1

(b) Option word 1 is the value in meters of the deflection bias which is to be defined as the minimum distance from the target column's path (a line) to the footprint centroid (a point).

The default value is 0.

Note: The side effects of deflection bias are discussed in section 4.8.

4.4.38 TARGET HEADING.

(a) general form: TARGET HEADING, option word 1 or TGTHDG, option word 1

(b) Option word 1 is a number whose value is the angle (in degrees) between the target's line of travel and the target-to-gun line.

The default value is 0.

Note: The side effects of target heading are discussed in section 4.8.

4.4.39 SEEKER SENSITIVITY.

(a) general form: SEEKER SENSITIVITY, option word 1 or SEEKSENS, option word 1

(b) Option word 1 is a number whose value is the minimum amount (threshold value) of energy per area (in joules per square meter) that must reach the COPPERHEAD seeker in order to initiate maneuver.

The default value is 60 joules/$m^2$. (The true value is not used as the default to avoid any possible classification problem.)

Note: The side effects of the seeker sensitivity value are discussed in section 4.8.

4.4.40 PROBABILITY OF A CORRECT MESSAGE.

(a) general form: PROBABILITY OF A CORRECT MESSAGE, option word 1 or PROBCORMSG, option word 1

(b) Option word 1 is a number whose value is the probability that the D.O. sends correct information to the FDC and the FDC interprets it correctly.

The default value is .975.

4.4.41 PROBABILITY DO WARNED TO LASE.

(a) general form: PROBABILITY DO WARNED TO LASE, option word 1 or PROBDOWARN, option word 1

(b) Option word 1 is a number whose value is the probability that the message warning the D.O. to begin lasing for the first COPPERHEAD round is transmitted and received successfully.

The default value is .975.

4.5 Special keywords: OVERRIDE, TEMPORARY, RESET.

4.5.1 Correspondence of Data Blocks to Keywords. As mentioned previously, a large part of the data used in each case is grouped into eight blocks. Each of these blocks is normally filled by reading a record from a word addressible mass storage file that is created by the PREPMS and PAM programs. The choice of which particular record to read from the file to fill each data block depends on the various keywords and option words for the case.

This correspondence of data blocks to keywords can be summarized as follows:

| Record to be read for data block of this type | Depends on options for this (these) keywords |
|---|---|
| (1) Weather | WEATHER |
| (2) Acquisition range data | ACQRNGDIST |
| (3) Response time data | RESPTIME |
| (4) Direct fire suppression data | DFIRESUPPR |
| (5) Random occurrence data | ACQRNGDIST and TGTVEL |
| (6) Probability of engagement data | RESPTIME, MSNCODE, DESIGTYPE, TGTVEL, GTRNG, REFLECT, ANGLE T, DEFLB, TGTHDG, SEEKSENS |

(7) Target posture distribution FO POSITION

(8) Invariant data INVARIANT

As can be seen, there is but one keyword influencing the choice of record to be read for each data block except in the cases of random occurrence data and probability of engagement data.

4.5.2 OVERRIDE and TEMPORARY Features. Two additional methods of filling in a data block are provided. The first is to choose a record from the word addressible mass storage file to be read into the data block without regard to the options chosen for the keyword(s) that would normally determine the record choice; this is the OVERRIDE feature. The second is to actually supply alternative data as traditional formatted FORTRAN inputs to be read into the data block; this is the TEMPORARY feature.

The OVERRIDE option is provided primarily so that one can choose to use probability of engagement (PE) data that has been generated with one set of conditions under other, not too different, conditions without having to re-run the PAM program. For example, PE data for a case with nominal response time, 130 sec. could be used to run a case identical (in those factors affecting PE) except for a nominal response time of 150 sec. The advantage of this procedure is that one can avoid re-running the relatively expensive PAM program in those cases where the factors affecting PE have changed very little. The disadvantage, of course, is that there are no efficient objective criteria available to determine whether the errors introduced by using wrong (but, it is hoped, approximate) PE data are significant in a given situation.

The TEMPORARY option is provided so that one can directly load data into any data block without having to use the PREPMS preprocessor program. This option allows one to use data for a given run without running the PREPMS program to make that data a "permanent" record in the data base file. This option was provided for flexibility, but was not exercised at all in the original COPPERHEAD Operational Performance Evaluation done for the COEA.

4.5.3 Using the OVERRIDE.

(a) general form: OVERRIDE , keyword, option word 1 or OVERIDE, keyword, option word 1

(b) Choices for keyword are:

(1) WEATHER or W

(2) ACQUISITION RANGE DISTRIBUTION or ACQRNGDIST

(3) RESPONSE TIME or RESPTIME

(4) DIRECT FIRE SUPPRESSION or DFIRESUPPR

(5) RANDOM OCCURRENCE DISTRIBUTION or RODIST

(6) PEDATA

(7) TARGET POSTURE DISTRIBUTION or TGTPSTDIST

(8) INVARIANT

Option word 1 is the name of any record (on the random access mass storage data base file) containing data for the data block type specified by the keyword choice.

(c) The OVERRIDE causes the selected data block (or blocks, if more than one OVERRIDE card is used in a case) to be filled with data from the record whose name is option word 1.

There is no default for OVERRIDE. An error will result from the omitting of either the keyword or option word 1.

Note: If the record name includes one or more blanks or separators then the record name must be preceded by a "$" (dollar sign). The ten characters forming the record name must immediately follow the dollar sign, any blanks in the name must be included in their proper positions, and no excess blanks may be inserted. For present purposes, separators include commas, slashes, question marks, and parentheses.

4.5.4 Using TEMPORARY.

(a) general form: TEMPORARY, keyword or TEMP, keyword

(b) Choices for keyword are as in section 4.5.3.

(c) The use of the TEMPORARY option will cause the program to begin reading data in FORTRAN formatted fashion. Hence, each use of the TEMPORARY option must be immediately followed by the formatted data cards to be read to fill the data block selected by the keyword.

The formats to be used and the data to be supplied for each block are identical to those used by the PREPMS program for the corresponding block (see section 14.4.2).

Note: If the keyword is PEDATA, then one has the further option of reading the formatted cards from a separate file (TAPE 4). To use this option, a card: USE TAPE 4 must be the next card after the TEMPORARY, PEDATA card; then the formatted data should be on a file to be read by I/O unit 4.

There is no default for TEMPORARY.

4.5.5 RESET.

(a) general form: RESET, keyword

(b) Allowable keywords are as in section 4.5.3.

(c) This card will reset the default conditions for the keyword chosen. Its use is mainly to undo an OVERRIDE or a TEMPORARY without having to use NEXT CASE, RD which might reset other defaults not desired.

Note: The effect of an OVERRIDE or TEMPORARY used on PEDATA or RANDOM OCCURRENCE DISTRIBUTION (RODIST) can be undone only by a RESET, a NEXT CASE, RD, or another OVERRIDE or TEMPORARY for the same data block.

The effect of an OVERRIDE or TEMPORARY for any other data block can also be undone by a card of the usual form for that data block's keyword (i.e., keyword, option word 1, option word 2, -----).

There is no default on the RESET card.

## 4.6 Comments.

4.6.1 General Remarks on Comments. The program allows three types of user comment cards to be included in the input. In addition, the program itself generates some comments under certain conditions. The program generated comments are detailed in sections 7.1 and 7.2.

User comments in the input card deck (or card image file) are identified by having "$" (dollar sign) in the first one, two, or three columns of the card.

4.6.2 $ Comments. Cards having a "$" in column 1 but not in column 2 are user comment cards which are totally ignored by the program. When one is read, the program merely goes to the next input card and reads that.

This type of comment is provided only so that the user can leave reminders, separate sections of input, or otherwise label inputs without having an effect on program printouts.

4.6.3 $$ Comments. Cards having "$" in columns 1 and 2 but not in column 3 are user comment cards which are reproduced within the current case's printout only.

The comment (contents of the card) will be reproduced (less the first two columns) with the symbols "** USER **" preceding it in the comment section of the case's printout. (See sample case, Appendix D .)

These comments should be used to note any special details about a case.

4.6.4 $$$ Comments. Cards having "$" in columns 1, 2, and 3 are user comments that are printed out immediately upon being encountered by the program. Again the symbols "** USER **" are appended to the front of the comment and the $'s in columns 1-3 are removed.

These comments are printed on a special page separate from any case's printout. It was originally intended that "$$$" comments be put at the beginning of the input before the FIRST CASE card; however, wherever they are put a special page will be printed containing them, and this page will occur in the printout immediately after the previous case's printout; or, (if there has been no case run yet) immediately after the title page.

## 4.7 DATA Statement Data.

There are many data which are set in data statements rather than read in from cards or disk file with each run or case.

As a general rule, these data are either labels (alphanumeric data) or numbers that do not change from run to run.

All of these data statements have been collected into the BLOCK DATA subprogram BDATA1. They are now described common block by common block:

4.7.1 ABRLBL. This common block contains the abort labels (i.e., alphanumeric texts) that identify the abort causes on the printouts. They reside in the IDABRT array and are entered as Hollerith constants.

4.7.2 ACHAR. This common block consists of the CHAR array which contains the keywords used to input COPE case descriptors. Each keyword is entered as a string of Hollerith constants. In addition, corresponding to each keyword character string are two numbers, the first of which is the keyword number and the second of which is the number of distinct option word character strings allowed with that keyword. (Note: The second number as entered in the data statement line for each keyword is overwritten by numbers taken from the record named OPTNNUMS on the random access mass storage data base file. Thus, when the number of distinct optionword character strings allowed with a given keyword changes, there is no need to adjust the data statement; the pre-processor program PREPMS is used to adjust the value in OPTNNUMS to the correct value. (See section 14.5.)

Also, NREC, the total number of keyword character strings, is set in this common block's data statements.

4.7.3 DESRNG. This common block contains the DESRNG array which consists of the default values for maximum designator range.

DESRNG (I, J) is the maximum designator range for designator type I under condition J where J=1 for day and J=2 for night.

4.7.4 DISPLY. This common block contains the DISPT, DISPM, RECLBL, and TMPLBL arrays, all of which are defined in the variable glossary (see chapter 12). Each of these arrays consists of alphanumeric constants entered as Hollerith data.

4.7.5 FLTTIM. This common block consists of the TOFARY array. The array gives the default times of flight: TOFARY (I) is the COPPERHEAD time of flight for a gun-to-target range of I (currently only I=8 and I=12 are used).

4.7.6 FSRESP. This common block consists of the FSRT array which contains the FT Sill supplied response time distribution.

4.7.7 HEADNG. This common block contains the ACQLBL, DAYLBL, DFSLBL, DGTLBL, DOLBL, ROLBL, and RSPLBL arrays all of which are defined in the glossary of variables (chapter 12).

Each of these arrays consists of alphanumeric constants entered as Hollerith data.

4.7.8 LOGFLG. This common block contains various logical flags: FIRSTL, SEQNML, SPCL array, and SHRTEC all of which are defined in the glossary of variables (chapter 12).

These are all logical type variables.

4.7.9 POINT. This common block contains the IPOINT array used in Subroutine INPUT to direct control to various parts of a computed GOTO.

4.7.10 SMOKED. Only the SMK2 and SMK5 arrays plus the variable FRONT of this common block are filled in via data statements. They are all defined in the variable glossary (chapter 12).

4.7.11 STITLE. This common block contains SLVERS and the VERDAT array. Both are used in the title page printout.

4.7.12 SYMBOL. This common block consists of the ALFBET, ANUMBR, and SEP arrays plus the single variables BLANK, DECPNT, and DOLLAR. They are all alphanumeric constants filled in as Hollerith data and defined in the glossary of variables (chapter 12).

4.7.13 XVALUE. This common block contains the XVALUE array which is used to generate PEDATA record names. (See sections 5.4 and 9.34.)

## 4.8 Side Effects of Certain Inputs.

There are a number of dependencies and side effects connected with some of the inputs.

First, the choice of response time data (4.4.15) and mission code (4.4.8) together with time-of-flight determine the nominal response time. (Time-of-flight itself may be determined either by direct input (4.4.24) or as a result of the choice of gun-to-target range (4.4.35)).

The nominal response time, designator type (4.4.11), target velocity (4.4.21), angle T (4.4.25), gun-to-target range (4.4.35), reflectivity (4.4.36), deflection bias (4.4.37), target heading (4.4.38), and seeker sensitivity (4.4.39) are all used to determine the name of the PE data record (see section 5.4). Indeed, angle T, reflectivity, deflection bias, target heading, and seeker sensitivity play no other roles in COPE than to affect the PE data (by determining the PE data record name) and to appear on the case heading page (see section 6.3) for identification purposes.

The maximum designation range (4.4.22) depends on the time of day (day or night) input with response time data (4.4.15) and the designator type (4.4.11). The maximum designator range can also be set by direct input (4.4.22); however, the user must take care to see that the PE data as well as TTF and RNGTTF arrays contain values appropriate for the maximum designator range used.

The target velocity (4.4.21) can be input directly or as a result of the time of day (day or night) used for the response time data (4.4.15), the month used for weather (4.4.13), and the terrain type (4.4.14) used for acquisition data. (The VELTBL array may need adjusting if additional choices are allowed for acquisition data or weather.)

Finally, the weather data (4.4.13) has a strong side effect on the effectiveness of smoke when the SMOKE option (4.4.19) that plays a specified number of smoke rounds is used.

CHAPTER 5

# 5. PREPARING TO RUN COPE

Prior to running the main COPE program, it is necessary to run the preprocessor programs to create the word addressable mass storage data base file (TAPE 11). This chapter describes the role of each preprocessor, the structure of TAPE 11, and the interfacing of TAPE 11 with each of the programs by means of the record names.

## 5.1 Preprocessor Programs.

There are three programs that may be regarded as preprocessors to the main COPE simulation program. These three programs are called:

(1) PREPMS which stands for preprocessor mass storage,

(2) PRBLOS which stands for probability of line-of-sight, and

(3) PAM which stands for probability of seeker acquisition and round maneuver.

More detailed information on PREPMS, PRBLOS, and PAM is included in chapters 14, 15, and 16 respectively of this report. In addition, more thorough documentation of PAM is to be published in a future separate AMSAA Technical Report.

As mentioned in chapter 4 of this report, much of the data used in each COPE case is grouped into eight data blocks. These data blocks are filled by reading records from TAPE 11 (except in the case of the TEMPORARY option described in section 4.5). The main purpose of the PREPMS program (and one of the purposes of the PAM program) is to fill values into the records on TAPE 11 in such a way that they are accessible to the main COPE program.

The PREPMS program is used to fill in records for data blocks for weather data, acquisition range data, response time data, direct fire suppression data, random occurrence data, target posture data, and invariant data. It can also be used to fill in records for PE (probability of engageme data, but this feature has never been used because the PAM program which generates PE data has been modified to write its results directly to TAPE 11 and the choice of record name has been entirely automated in this case. Finally, the PREPMS program is also used to establish and modify the option words allowed as choices with each keyword (described in sections 4.4.1 through 4.4.19 except for those optionword choices described as "a numerical value" in sections 4.4.18 and 4.4.19).

The PRBLOS program is used to calculate the probabilities of line-of-sight at various ranges. These values are then used in the random occurrence line-of-sight model. They are read from the PRBLOS printout and entered by hand as input to PREPMS. Because the PRBLOS program does not directly act on TAPE 11, it shall not concern us further in this chapter (see chapter 15 for complete discussion of PRBLOS).

The PAM program takes as input the parameter values describing the conditions under which probability of engagement is to be computed. It then computes the PE data values, creates a record name for its current block of PE data by encoding the input conditions according to a scheme also used in the main COPE program, and then writes the PE data to TAPE 11 under that record name.

## 5.2 Word Addressable Mass Storage Data Base File (TAPE 11).

The word addressable mass storage data base file (TAPE 11) consists of the records containing the data to be used by each of the eight data blocks mentioned in chapter 4, the records of option words allowed with each keyword (applies only to keywords described in sections 4.4.1 through 4.4.19 and excludes any option words described as "a numerical value"), and a record called OPTNNUMS that records the number of option word character strings allowed for each keyword.

TAPE 11 is a CDC word addressable mass storage file using a name type master index. (This file type and the way it is interfaced with a FORTRAN program is explained in the CDC Fortran Extended Version 4 Reference Manual chapter 8, a section titled "Mass Storage Input/Output"). The important feature to note here is that this file type (with a name type master index) is organized as a set of records each consisting of one or more words of data and each having a unique record name of one to ten characters. This unique record name is set when the record is created and must be used by any program that either reads or modifies the record. It is by using the same system of record names in the COPE program as in the preprocessors PREPPE and PREPMS that the data records written by the preprocessors can be read by the main COPE program.

## 5.3 Record Names.

### 5.3.1 Weather Data Record Name.

The record name of the data to be used in the weather data block (the WDATA array) has the form : mmmtttt021 where mmm are the first three letters of the name of the month of the weather data, tttt are four numbers giving the time of day (24 hour military time) of the weather data, and the 021 are three numbers signifying that WEATHER has keyword number 21.

Examples:

JUN0600021 is the record name for weather data for a June day at 0600 (6 AM).

SEP2200021 is the record name for weather data for a September day at 2200 (10 PM).

5.3.2 Acquisition Data Record Name. The record name of the data to be used in the acquisition range data block (the ACQDAT array) has the form: ACQDATnn22 where nn are two digits giving the number of the choice for option word 1 used with keyword 22 (section 4.4.14).

Examples:

ACQDAT0122 is the record name of the acquisition range and LOS segment length data to be used when option word AVERAGE is used with keyword 22 (ACQRNGDIST)

ACQDAT0322 is the record name of the data to be used with OPEN terrain.

5.3.3 Response Time Data Record Name. The record name of the data to be used in the response time data block (RSPDAT array) is of the form: RSPDATnn23 where nn are two digits giving the number of the choice for option word 1 used with keyword 23 (section 4.4.15)

Examples:

RSPDAT0123 is the name of the record containing the response time data to be used when the FTSILL option is selected.

RSPDAT0223 is the name of the record containing the response time data for the STAUCH option.

Note: There is no data record corresponding to the PARAM choice for option word 1. This is because the PARAM choice for option word 1 causes the program not to require any of the data in the response time data block.

5.3 4 Direct Fire Suppression Data Record Name. The record name of the data to be used in the direct fire suppression data block (DFSDAT array) is of the form: DFSDATnn24 where nn are two digits giving the number of the choice for option word 1 used with keyword 24 (section 4.4.16)

Example:

DFSDAT0124 is the record name for HIGH direct fire suppression data.

5.3.5 Random Occurrence Data Record Name. The record name of the data to be used in the random occurrence data block (RODATA array) is of the form: RODAVmTRNn where m is the target velocity in meters per second (obviously between 0 and 9) and n is the terrain type number (corresponds

to number of option word 1 on ACQRNGDIST card, i.e., 1 for AVERAGE, 2 for CLOSE, 3 for OPEN).

Example:

RODAV8TRN2 is the record name for the random occurrence data to be used when target velocity is 8 m/s and terrain type is 2 (CLOSE).

Note: The program is currently set up only for velocities of 2, 3, 5, 8, and 9. Furthermore, 2 and 8 are treated as 3 and 9 respectively when it comes to looking up RODATA and PEDATA.

5.3.6 Target Posture Distribution Data Record Name. The record name of the data to be used in the target posture data block (PSTDAT array) is of the form: PSTDATnn27 where nn is the number of the choice for option word 1 used with keyword 17 (section 4.4.12).

Example:

PSTDAT0127 is the record name for the target posture distribution used when FO POSITION is VANTAGE POINT.

Note: Keyword 17 (FO POSITION) and keyword 27 (TARGET POSTURE DISTRIBUTION) could each be used to control the choice of target posture distribution data, but only keyword 17 (FO POSITION) has been provided with optionwords and is the only one of the two explained in section 4.4.

5.3.7 Invariant Data Record Name. The record name of the data used in the invariant data block (AINVDA array) is of the form: AINVDAnn28 where nn is the number of the choice for option word 1 used with keyword 28.

Example:

AINVDA0128 is the record name for the invariant data used with the default (omitted) option word 1.

Note: The keyword INVARIANT is currently set up to accept no option words (other than a blank which is equivalent to the default).

5.3.8 Option Word Record Names. For each keyword numbered less than 41 (see section 5.3.9 for keyword numbers) there may be some option words that are not numerical values (see sections 4.4.1 through 4.4.19 for the current option word choices available with each keyword).

To create, modify, or add to the list of character strings that can be used with any keyword, it is necessary to run PREPMS with input specifying the keyword (in its abbreviated form of ten characters or less), the total number of option word character strings that can be used

with that keyword and the keyword number as well as the option word number, the option word choice number and the option word character string for each option word choice. The formatting of this data is explained in chapter 14, but an explanation of its meaning is provided by the sample input following:

```
OPTION WORDS, W, 10, 21
1, 2, JUNE
1, 2, J
2, 2, DECEMBER
2, 2, D
1, 3, 1400
1, 3, 14
2, 3, 0600
2, 3, 06
3, 3, 2200
3, 3, 22
```

where the first line indicates that option word allowable choices are being set for keyword W (or WEATHER), that ten optionword character strings are being input, and that W (or WEATHER) is keyword number 21.

Each of the next ten lines establishes a character string as a possible option word choice. For example, 1, 3, 1400 establishes 1400 as a character string corresponding to the first choice for option word 2 of any input line beginning with keyword W (or WEATHER). The first digit (1) indicates that 1400 is the first choice allowed, the second digit 3 indicates that 1400 is a choice allowed for the third item on the input line (recall that an input line is of the form: Keyword, option word 1, option word 2, - - -, option word n so that option word 2 is actually the third item on an input line), and, of course, 1400 indicates that the input character string to select option 1 for option word 2 is 1400. The fact that the next line is 1, 3, 14 means that 14 has the same effect (first choice for option word 2) as 1400. No more than two lines are allowed to be input with the same ordered pair (choice number, option word number + 1) under any given keyword (though one may input just one line for such a pair if desired).

The PREPMS program then places this option word information on a record whose record name is the keyword (W in the case above). It is for this reason that the keyword used must be ten or fewer characters. Each keyword of length more than ten characters also has an abbreviated form (see the appropriate section of chapter 4 for each keyword; even some keywords of length less than ten characters have abbreviated forms). For example, the keyword ACQUISITION RANGE DISTRIBUTION is suitable for input to COPE, but cannot be used as input to PREPMS because it is too long to be a record name; instead, the abbreviated form ACQRNGDIST must be used when setting option words via PREPMS.

The rule is always to use the abbreviated form of the keyword with PREPMS input (even if the long form of the keyword is itself ten or fewer characters) and to use either form with input to COPE. For keywords with only one form, use it as input to both programs.

Finally, for each new option word choice that is allowed, the user must consider what COPE will do when it encounters the new option word. In the case of option words that control the choice of records to be read from TAPE 11 into data blocks, the user must also have run PREPMS to add the possible data block records into TAPE 11. In our example above, input to PREPMS must have created records named JUN0600021, JUN1400021, JUN2200021, DEC0600021, DEC1400021, and DEC2200021 containing the weather data to be used with each of the possible option word combinations. For the case of option words that control program actions rather than select a data record name, it will usually be necessary to modify the program code so that it will "know" what to do when encountering a new option.

5.3.9 Keyword Numbers, Abbreviated Forms, Etc. The following table gives the current list of allowed keywords, each keyword's number, abbreviated form, and relevant notes. (This overlays some of the information in chapter 4, but it was felt desirable to have all the keywords listed in one place for quick reference; chapter 4, sections 4.4.1 - 4.4.38 are still essential for allowed optionwords and detailed effects of keywords). When two consecutive keywords have the same number, the second is the abbreviated form. (Recall that blanks may be arbitrarily inserted in (or removed from) keywords.) Notes are explained at end of listing.

| Keyword Number | Keyword | Notes |
|---|---|---|
| 1 | FIRST CASE | *1 Sec. 4.4.1 |
| 2 | NEXT CASE | *2 Sec. 4.4.2 |
| 3 | END OF CASE | *1 Sec. 4.4.3 |
| 3 | END | *1 " |
| 4 | END OF FINAL CASE | *1 Sec. 4.4.4 |
| 4 | END F | *1 " |
| 5 | PLOT | *2 Sec. 4.4.5 |
| 6 | OVERRIDE | *3 Sec. 4.5.3 |
| 6 | OVERIDE | *3 " |
| 7 | TEMPORARY | *3 Sec. 4.5.4 |
| 7 | TEMP | *3 " |

| Keyword Number | Keyword | Notes |
|---|---|---|
| 8 | RESET | *3 Sec. 4.5.5 |
| 9 | CONTROL | *2 Sec. 4.4.6 |
| 11 | LINE OF SIGHT MODE | *2 Sec. 4.4.7 |
| 11 | LOS MODE | *2 " |
| 12 | MISSION CODE | *2, *9 Sec. 4.4.8 |
| 12 | MSN CODE | *2, *9 " |
| 13 | NUMBER OF DIFFERENT TARGET POSTURES | *2 Sec. 4.4.9 |
| 13 | NUMDIFTGTP | *2 Sec. 4.4.9 |
| 14 | TARGET TYPE | *2 Sec. 4.4.10 |
| 16 | DESIGNATOR TYPE | *2, *9 Sec. 4.4.11 |
| 16 | DESIGTYPE | *2, *9 " |
| 17 | FO POSITION | *4 Sec. 4.4.12 |
| 21 | WEATHER | *4 Sec. 4.4.13 |
| 21 | W | *4 " |
| 22 | ACQUISITION RANGE DISTRIBUTION | *4 Sec. 4.4.14 |
| 22 | ACQRNGDIST | *4 " |
| 23 | RESPONSE TIME | *2, *4, *9 Sec. 4.4.15 |
| 23 | RESPTIME | *2, *4, *9 " |
| 24 | DIRECT FIRE SUPPRESSION | *4 Sec. 4.4.16 |
| 24 | DFIRESUPPR | *4 " |
| 25 | RANDOM OCCURRENCE DISTRIBUTION | *10 |
| 25 | RODIST | *10 |
| 26 | PEDATA | *10 |
| 27 | TARGET POSTURE DISTRIBUTION | *10 |

| Keyword Number | Keyword | Notes |
|---|---|---|
| 27 | TGTPSTDIST | *10 |
| 28 | INVARIANT | *5 Sec. 4.4.17 |
| 31 | RUMC | *6 Sec. 4.4.18 |
| 32 | SMOKE | *7 Sec. 4.4.19 |
| 41 | DO ARTILLERY PK | *8 Sec. 4.4.20 |
| 41 | DOARPK | *8 " |
| 42 | TARGET VELOCITY | *9 Sec. 4.4.21 |
| 42 | TGTVEL | *9 Sec. 4.4.21 |
| 43 | MAXIMUM DESIGNATOR RANGE | *8 Sec. 4.4.22 |
| 43 | MAXDESRNG | *8 " |
| 44 | BAIL OUT RANGE | *8 Sec. 4.4.23 |
| 44 | BAIL OUTRNG | *8 " |
| 45 | TIME OF FLIGHT | *8 Sec. 4.4.24 |
| 45 | TOF | *8 " |
| 46 | ANGLE T | *9 Sec. 4.4.25 |
| 47 | DUST | *8 Sec. 4.4.26 |
| 48 | PROBABILITY OF SUCCESSFUL VOICE TRANSMISSION | *8 Sec. 4.4.27 |
| 48 | PRBVOCTRAN | *8 " |
| 49 | PROBABILITY OF SUCCESSFUL DIGITAL TRANSMISSION | *8 Sec. 4.4.28 |
| 49 | PRBDGTTRAN | *8 " |
| 50 | ROUND IN FLIGHT RELIABILITY | *8 Sec. 4.4.29 |
| 50 | RNDFLTREL | *8 " |
| 51 | TIME BETWEEN ROUNDS | *8 Sec. 4.4.30 |

| Keyword Number | Keyword | Notes |
|---|---|---|
| 51 | TBR | *8 Sec. 4.4.30 |
| 52 | NUMBER OF REPLICATIONS | *8 Sec. 4.4.31 |
| 52 | NUMREP | *8 " |
| 53 | NUMBER OF ROUNDS TO BE FIRED | *8 Sec. 4.4.32 |
| 53 | NUMRNDS | *8 " |
| 54 | NUMBER OF VEHICLES PER TARGET | *8 Sec. 4.4.33 |
| 54 | NUMVEHTGT | *8 " |
| 55 | DISTANCE BETWEEN VEHICLES | *8 Sec. 4.4.34 |
| 55 | DISTBVEH | *8 " |
| 56 | GUN TARGET RANGE | *9 Sec. 4.4.35 |
| 56 | GTRNG | *9 " |
| 57 | REFLECTIVITY | *9 Sec. 4.4.36 |
| 57 | REFLECT | *9 " |
| 58 | DEFLECTION BIAS | *9 Sec. 4.4.37 |
| 58 | DEFLB | *9 " |
| 59 | TARGET HEADING | *9 Sec. 4.4.38 |
| 59 | TGTHDG | *9 " |
| 60 | SEEKER SENSITIVITY | *9 Sec. 4.4.39 |
| 60 | SEEKSENS | *9 " |
| 61 | PROBABI[illegible] CORRECT MESSAGE | *8 Sec. 4.4.40 |
| 61 | PROBCORMSG | *8 " |
| 62 | PROBABILITY DO WARNED TO LASE | *8 Sec. 4.4.41 |
| 62 | PROBDOWARN | *8 " |

Explanation of notes:

*1 These keywords do not take option words.

*2 These keywords set certain flags or indices that cause the program to function in one of several alternate ways or else serve as indices for array "look-ups." Any increase in the number of options corresponding to one of these keywords might require code changes to COPE (at the least, redimensioning of some arrays) to specify the actions to be taken for the new options.

*3 These keywords take other keywords (numbers 21 through 28) as their second fields (i.e., as the first item following the separator after the keyword). The items to be used in the second fields are built into the program in data statements (CHAR array) and any changes would require code modifications to COPE. (See sections 4.5.1 through 4.5.5)

*4 These keywords set the names of records on TAPE 11 to be read into the program data blocks. Any increase in the number of options allowed must be accompanied by input to PREPMS creating the records with the names that would be called for if the new options were exercised. (See chapter 14.)

Also, keyword 23 takes some further option words which determine digital or voice communications and day or night visibility. If more options are allowed for these keywords, then code changes are required for COPE.

*5 Currently takes no option words, but if option words were added, it would fall in the same class as keywords with note *4.

*6 The addition of any new options for RUMC would require code changes to COPE.

*7 The addition of any new options for SMOKE would require code changes to COPE.

*8 These keywords all take numerical values for option word 1. Any numerical value that could be read in F10.4 format is allowable though for some option words negative values or values greater than 1.0 may cause program malfunction (as in the case when the value represents a probability). Any attempt to modify one of these keywords to accept options other than numbers would require program code changes in COPE.

*9 These keywords affect the choice of the PEDATA record name to be used (see section 5.4). Any increase in the number of allowed choices here would require changing values in the XVALUE array (in both COPE and PAM) and then running PAM to create any new PEDATA records that may be called for as a result of these changes.

In short, one can see from the above notes that while it is easy to add new optionword choices for a particular keyword (by running the PREPMS program when necessary), it often requires code changes to COPE to make the new choices have the desired effect. About the only keywords for which options can be conveniently changed are those of notes 4, 5, and, to a lesser extent, 9.

## 5.4 PEDATA Record Names.

The record name for the PEDATA (probability of engagement data) to be used with a given case is a function of ten keywords and their associated optionwords (except, of course, when using the OVERRIDE or TEMPORARY keywords with PEDATA).

When the PAM program is run, nine parameters are input. The values of these parameters determine the record name of the PEDATA created by that run. When COPE is run, the same nine parameters are assigned values (as the result of keyword inputs or defaults), the same function for determining record name is applied, and the record with the resulting name is read from TAPE 11 and filled into the PEDATA data block.

Because there are only a finite number of ten character record names and because the record name depends on the values of nine different parameters, it is necessary to limit the choices allowed for each parameter. This section is devoted mainly to a description of the function that determines record name when given the nine parameter values and to the changes required to introduce a new allowed parameter value.

The nine parameters are:

(1) nominal response time in seconds (determined by keywords 23 and 12),

(2) designator type (determined by keyword 16),

(3) target velocity (determined by keyword 42),

(4) gun-to-target range (determined by keyword 56),

(5) target reflectivity (determined by keyword 57),

(6) angle T (i.e., angle between gun-target line and designator target line; determined by keyword 46),

(7) deflection bias (determined by keyword 58),

(8) target heading (determined by keyword 59),

(9) seeker sensitivity (determined by keyword 60).

The values currently allowed are (by parameter number):

(1) any time t in seconds such that 0.5 second $\leq$ t <999.5 seconds

(2) GLLD, MULE, LTD (which COPE and PAM call types 1, 2, and 3 respectively)

(3) 2, 3, 5, 8, and 9 meters per second,

(4) 8 and 12 kilometers,

(5) .05, .10, .20, and .30,

(6) 0, 25, 30, 60, 90, and 120 degrees,

(7) -200, -100, 0, 100, and 200 meters,

(8) -60, -30, 0, 30, and 60 degrees,

(9) 24, 36, 48, 60, and 72 joules per square meter.

The record name function works as follows:

-First, it rounds the nominal response time (parameter one to the nearest integer (call the result t', then $1 \leq t' \leq 999$).

-Next, for each parameter 2 through 9, the function matches the parameter value to one of the allowed values. Say for the jth parameter, the value is found to match the Ijth allowed value (if no match is found, an error stop results).

-Next, the record name is made by converting t' to octal (call it t" where $1_8 \leq t'' \leq 1747_8$), by letting $N_i$ = a function of $I_j$ (usually $I_j$-1), and then defining

record name = $20053333xxxxN_2N_3N_4N_5N_6N_7N_8N_9$ in octal

where: 20053333 is octal for the alphanumeric "PE00", xxxx are the four octal digits of t" (possibly with leading zero(s) if t" is small enough), $N_2$,_ _ _, $N_9$ are as defined above (note that when $1 \leq I_j \leq 8$, we have $0 \leq N_i \leq 7$ so that each $N_i$ is but a single octal digit. It is for this reason that we shall never allow more than eight different choices for the value of any of the parameters 2 through 9).

The values allowed for each parameter 2 through 9 are to be found in the XVALUE array which is defined by data statements in both COPE and PAM (also, in PREPMS). In COPE, the data statement is in BDAT1, in PAM it is in subroutine PENAME, and in PREPMS it is in BDATA2.

The XVALUE array is set up so that XVALUE (I,J,K) is the Ith allowed value for the Jth parameter when K=1 and XVALUE (I,J,K) is the $N_J$ value (as defined above a function of $I_J$ with $0<N_J<7$) corresponding to the Ith allowed value of the Jth parameter when K=2.

Notes: (1) Because the first parameter (nominal response time) is allowed a wider range of values, the XVALUE entries with J=1 are not used. (2) Because XVALUE is dimensioned XVALUE (8,9,2) and because not all parameters 2 through 9 actually use the maximum of eight different allowable values, those array spaces XVALUE (I,J,1) and XVALUE (I,J,2) for I greater than the number of allowable values for parameter J are filled with 0's and 8's respectively. Hence, if an NJ value of 8 were to occur, the program has made an invalid parameter value match and a stop with error message will occur. (3) The programs do not distinguish between target velocities of 2 and 3 m/s or between 8 and 9 m/s or between designator types MULE and LTD when it comes to PE data. (There are still distinctions made in other parts of COPE such as calculating current designator target ranges or looking up LDWSS probability of hit numbers as function of designator type). This failure to distinguish can be observed in the XVALUE array: the XVALUE (I,J,2) values are identical for target velocities of 2 and 3 m/s, for example.

To allow more choices for one of the parameters 2 through 9, one need merely modify the data statements defining the XVALUE array so that XVALUE (I, J, 1) with appropriate I and J has the new value and XVALUE (I,J,2) has the corresponding NJ (octal digit). This must be done in both COPE and PAM (and should be done in PREPMS to keep it current also). Then one must run PAM with the parameter combinations using the new allowed value(s) to create the PEDATA corresponding to the case(s) one intends to run in COPE with the new parameter values.

To allow a new choice for parameter 1, one needs merely run PAM with the new value for parameter 1 to create the PEDATA for the case(s) using the new nominal response time and the desired allowed values for the other parameters. Of course, nominal response times outside .5 seconds to 999.5 seconds are not allowed.

Note: The PREPMS program uses the XVALUE array only to decipher PE record names. That is, one can direct that program to list the PEDATA records currently on TAPE 11 and then decipher the PE record names and print the values of the 9 parameters to which each such record corresponds. It is for this reason that the XVALUE array data statement in PREPMS should be updated along with those in COPE and PAM whenever a new value is introduced for any of the parameters 2 through 9.

## 5.5 Control Cards for COPE and Its Preprocessors.

The appendixes A, B, C, and D at the end of this report include respectively, the control cards used to run PAM, PPBLOS, PREPMS, and COPE. They are explained in these appendixes and hence require no further explanation here. These control card sets, of course, apply only to BRL's computer system and would require minor changes for other CDC installations and major changes for non-CDC computers.



Next page is blank.

# 6. OUTPUT

This chapter describes the normal COPE output prints. Special messages and diagnostic or error prints that may occur are described in Chapter 7.

The normal COPE output consists of four types of pages, three of which will always be present and one of which will be present only if one or more "$$$" comments is included in the input.

The page types will ordinarily occur in this order:

Title Page
"$$$" comment page (if present)
First case heading page
First case results page
Second case heading page
Second case results page

nth case heading page
nth case results page

final case heading page
final case results page

An additional "$$$" comment page will occur following the results page of the last case whose END card precedes any "$$$' comment card (or cards).

Each page type is now described in detail.

## 6.1 Title Page.

This page includes the program name, the names of the program author and the various contributors to the model design and data collection. In addition, it identifies the activity, division, branch and section responsible for the model.

Of greater practical value, this page also includes the version number of the program and the date that version was created. Finally, the page gives the date and time of the current run of the program.

For an example, see Listing 6-1.

## 6.2 "$$$" Comment Page.

This page merely lists the user's "$$$' comments (see section 4.6.4) with the symbol "** USER **" to the left of each comment line. In addition, any program generated comments (see section 7.1) are listed with "**COPE**" to the left of each comment line.

```
C O P P E R H E A D   O P E R A T I O N A L   P E R F O R M A N C E   E V A L U A T I O N


          -PROGRAM BY RICHARD SANDMEYER
          -MODEL DEVELOPMENT AND DATA COLLECTION BY:
            DAVID BARNHART        JULIAN CHERNICK      DIANA FREDERICK
            RICHARD SANDMEYER     RICHARD SCUNGIO      MICHAEL STARKS
            EDWARD STAUCH         ANNIE YOUNG
          -BASED ON A PROPOSAL BY HERBERT FALLIN
          -DONE AT THE REQUEST OF DAVID HARDISON (DUSA-OR)


               UNITED STATES ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY
                             GROUND WARFARE DIVISION
                          SUPPORT WEAPONS ANALYSIS BRANCH
                            SYSTEMS EVALUATION SECTION


                   CODE VERSION 6.2 CURRENT AS OF 6 MARCH 1980
                     THIS RUN DONE ON 08/14/80 AT TIME 15.41.16.
```

LISTING 6-1 TITLE PAGE

For an example, see Listing 6-2.

## 6.3 Case Heading Page.

The first page for each case is the heading page. The heading page gives a description of the conditions in effect for the case whose results follow. In addition, the heading page contains a section for any "$$" comments (either user or program generated). The case heading page (Listing 6-3) begins with a line indicating the case number. The next line below that gives the number of replications (i.e., number of potential Copperhead missions generated and tested by the current case of the Monte Carlo COPE model).

The next area of the case heading page is broken up into nine boxes each of which is surrounded by asterisks and contains descriptors of one aspect of the current case. These boxes are now described one at a time.

Note: When an OVERRIDE or TEMPORARY option is in effect, the case heading information may be unreliable for those items connected with the data block for which the OVERRIDE or TEMPORARY is in effect.

In the special case of a PE data block input via the OVERRIDE option, the values given in the comment of section 7.2.6 take precedent over those in the case heading when it comes to determining PE data but not for other purposes (for example, velocity used for calculating designator-to-target range is the velocity printed in the TARGET box rather than that in the comment).

If in doubt, the user should look at the contents (Data Check) of TAPE 8 for the case in question (see section 7.6) to determine which values were actually used for various parameters.

6.3.1 TARGET. The first box is labeled "TARGET" and contains data describing the target. These data are:

(a) Target type

(b) number of vehicles in the target unit,

(c) mean distance between target vehicles,

(d) target velocity (speed)

(e) target heading (where a heading of zero means the target is moving along the gun-to-target line toward the gun. Angles are measured counterclockwise from this zero heading).

(f) target reflectivity.

6.3.2 FIRING INFO. Moving vertically down the page, one next finds the FIRING INFO box which gives various information about the firing

```
$$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$     COMMENTS FOR THIS SET OF CASES    $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$
** USER **  THIS COMMENT WILL APPEAR ON THE $$$ COMMENT PAGE
```

LISTING 6-2   "$$$" COMMENT PAGE

```
* * * * * * * * * * * * * * * * * * * * * * * * *   C A S E   N U M B E R   1 * * * * * * * * * * * * * * * * * * * * * * * * *

NUMBER OF REPLICATIONS=10000
```

```
TARGET

TYPE  1
NUMBER OF VEHICLES=10
MEAN DISTANCE BETWEEN VEHICLES= 61.M
TARGET VELOCITY=3. M/SEC
TARGET HEADING=  0. DEGREES
TARGET'S REFLECTIVITY= .10
```

```
FIRING INFO

NUMBER OF ROUNDS PER ENGAGEMENT= 6
ROUND RELIABILITY= .96
TIME BETWEEN ROUNDS= 20. SEC
TIME OF FLIGHT= 31.0 SEC
NOMINAL GUN-TARGET RANGE=  8. KM
ANGLE T= 25. DEGREES
DEFLECTION BIAS=   0. METERS
```

```
MISC

DELTA T3 ASSUMED TO BE ZERO
SEEKER SENSITIVITY= 60. JOULES/M**2
```

```
DETECT, COMMO, AND PROC TIMES

PRIORITY PRE-PLANNED TARGET
DIGITAL COMMO
DAY TIME
NOMINAL RESPONSE TIME=106. SEC
PROB SUCCESSFUL DIGITAL TRAN= .915
PROB CORRECT MESSAGE =  .975
PROB DO WARNED TO LASE =  .975
```

```
DESIGNATOR

TYPE  1
MAXIMUM DESIGNATOR RANGE=7000. M
BAIL-OUT RANGE=1000. M
DESIGNATOR AT VANTAGE POINT
```

```
DESIGNATOR SUPPRESSED

PROB SUPPRESSED BY INDIRECT FIRE=.01
PROB SUPPRESSED BY DIRECT FIRE: HIGH
```

```
WEATHER

MONTH OF JUN
TIME OF DAY: 0600
```

```
TERRAIN AND LOS

CLOSE TERRAIN
CONTINUOUS LOS ("SHOOTING GALLERY")
```

```
OBSCURANTS

PROB SMOKE KILLS MISSION= .75
VOLUME OF SMOKE FIRED:  484 +  846
PROB DUST KILLS MISSION= .37
```

```
$$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$      C O M M E N T S      $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$

** CDPE **   IN PPLOT FOR CASE  1 WITH IPPFG= 1 .  NO PLOT PRODUCED .
```

LISTING 6-3   CASE HEADING PAGE

of the COPPERHEAD round. These items are:

(a) number of rounds fired per engagement,

(b) round in-flight reliability

(c) time between successive rounds

(d) time of flight of round

(e) nominal gun-to-target range

(f) angle T (angle between target-to-gun line and target-to-designator line)

(g) deflection bias (which is the minimum distance between the COPPERHEAD footprint centroid and the target's path).

6.3.3 MISC. The next box below the FIRING INFO is labeled MISC and contains miscellaneous data relevant to the case. These data are:

(a) either a note that DELTA T3 is zero (in which case RUMC (see section 4.4.18) is optimally chosen for each fire mission or a note that DELTA T3 is not assumed to be zero in which case RUMC is a fixed value for each fire mission.

(b) The value of RUMC (if DELTA T3 is not assumed zero; otherwise, this line is not printed.)

(c) The seeker sensitivity value used for the COPPERHEAD seeker in this case.

6.3.4 DETECT, COMMO, and PROC TIMES. The next box which is located at the top of the second column of boxes contains data pertaining to detection, communication, and processing times. These data are:

(a) The mission type

(b) The type of communication (i.e., digital or voice)

(c) day or night indicator

(d) nominal response time (which is the sum of the median communication and processing times and the time of flight)

(e) probability of successful digital communication link being established when needed (not printed when item (b) above is VOICE)

(f) probability of successful voice communication link being established when needed (printed only when applicable - see section 2.2.9)

(g) probability of a correct message,

(h) probability that the D.O. is warned to lase.

If the parameterized response time option is used, then the parameterized response time (which does not include time of flight) is listed in this box first.

6.3.5 DESIGNATOR. The next box gives information about the designator (both the soldier and the piece of equipment). This information consists of:

(a) designator type,

(b) maximum designator range,

(c) bail-out range

(d) designator location.

6.3.6 DESIGNATOR SUPPRESSED. The next box gives information about suppression of the designator. This information is:

(a) The probability that the FO (D.O.) is killed by Red preparatory artillery fires.

(b) A label giving the level of direct fire suppression achieved by the target against the D.O.

6.3.7 WEATHER. The first box of the third column gives weather data. These data are:

(a) The month whose weather is being used in the current case,

(b) The time of day (on a 0 to 2400 clock) whose weather is being used in the current case.

6.3.8 TERRAIN AND LOS. The next box moving down has terrain and line-of-sight information. This information includes:

(a) The terrain type used for the acquisition range distribution and line-of-sight segment length distribution.

(b) The line-of-sight model used in the current case ("SHOOTING GALLERY" or "RANDOM OCCURRENCE").

6.3.9 OBSCURANTS. The final box gives data on obscurants (smoke and dust). These data are:

(a) Probability that smoke is sufficiently dense to kill a potential COPPERHEAD mission.

(b) Number of smoke rounds fired of types one and two respectively to obtain the level of smoke in (a) above. (This is printed only if smoke effects are played as a function of number of smoke rounds fired (see section 4.4.19).

(c) Probability that dust from RED HE artillery fire is sufficient to abort a potential COPPERHEAD mission.

6.3.10 "$$" Comment Section.

The final section of the heading page for each case consists of a comment section. This comment section includes both any user comments for this case as well as any program generated comments.

If a case has no comments of either type, then the comment section is omitted.

6.4 Case Results Page.

Following the heading page for a given case is the case results page. This page is labeled in such a fashion as to be almost self-explanatory (see Listing 6-4).

The page has one line corresponding to each mission abort condition and two lines for each round abort condition. A mission abort occurs whenever one of the tests that affect the entire potential COPPERHEAD mission is failed (see sections 2.2.2 through 2.2.11); a round abort occurs whenever one of the tests that affect the functioning of a single round is failed (see sections 2.2.12-2.2.20).

Beginning at the top of the results page, one finds the number of occasions, which is the number of potential COPPERHEAD missions sampled for the current case. This number should equal the number of replications on the heading page for the same case.

The first seven abort causes (or tests) are those which would prevent the D.O. from even attempting to employ COPPERHEAD against a target.

Each of these causes of abort has one line giving the following information:

(a) The number of the abort cause (or test),

(b) a label describing the cause of abort,

```
           CAUSE OF ABORT
                                        ***** OCCASIONS (10000) *****
  1   DESIGNATOR IN INDIRECT FIRE        84(0.84%)   NUMBER TESTED (% PASSED):  10000(99.16%)
  2   TARGET BEYOND VISIBILITY RANGE    415(4.15%)   NUMBER TESTED (% PASSED):   9916(95.81%)
  3   TARGET BEYOND DESIGNATOR RANGE      0(0.00%)   NUMBER TESTED (% PASSED):   9501(100.0%)
  4   TARGET NOT DETECTED                 0(0.00%)   NUMBER TESTED (% PASSED):   9501(100.0%)
  5   SMOKE KILLED MISSION             7123(71.23%)  NUMBER TESTED (% PASSED):   9501(25.03%)
  6   DUST KILLED MISSION               887(8.87%)   NUMBER TESTED (% PASSED):   2378(62.70%)
  7   DESIGNATOR BAILED-OUT PRE-COMM    484(4.84%)   NUMBER TESTED (% PASSED):   1491(67.54%)
                                   ***** ATTEMPTED ENGAGEMENTS ( 1007) *****
  8   COMMO OUT (BOTH VOICE & DIGTL)     70(0.70%)   NUMBER TESTED (% PASSED):   1007(93.05%)
  9   ERROR IN TRANSMISSION              21(0.21%)   NUMBER TESTED (% PASSED):    937(97.76%)
 10   DESIGNATOR BAILED-OUT POST-COM    159(1.59%)   NUMBER TESTED (% PASSED):    916(82.64%)
 11   LOS LOST - NO FIRING               55(0.55%)   NUMBER TESTED (% PASSED):    757(92.73%)
                                          ***** SHOTS (  702) *****
        ROUND NUMBER :                 1            2            3            4            5            6

 12   LOS LOST DURING MISSION        30(0.30%)    62(0.62%)    96(0.96%)   151(1.51%)   242(2.42%)   345(3.45%)
        NUMBER TESTED (% PASSED)    702(95.73%)  702(91.17%)  702(86.32%)  702(78.49%)  702(65.53%)  702(50.85%)

 13   DESIGNATOR NOT WARNED IN TIME  15(0.15%)     0(0.00%)     0(0.00%)     0(0.00%)     0(0.00%)     0(0.00%)
        NUMBER TESTED (% PASSED)    672(97.77%)  640(100.0%)  606(100.0%)  551(100.0%)  460(100.0%)  357(100.0%)

 14   DESIGNATOR IN DIRECT FIRE     176(1.76%)   173(1.73%)   194(1.94%)   183(1.83%)   164(1.64%)   144(1.44%)
        NUMBER TESTED (% PASSED)    657(73.21%)  640(72.97%)  606(67.99%)  551(66.79%)  460(64.35%)  357(59.66%)

 15   ROUND RELIABILITY FAILURE      20(0.20%)    16(0.16%)    15(0.15%)    16(0.16%)     9(0.09%)    13(0.13%)
        NUMBER TESTED (% PASSED)    481(95.84%)  467(96.57%)  412(96.36%)  368(95.65%)  296(96.96%)  213(93.90%)

 16   TARGET OBSCURED BY MINI-TERRN  54(0.54%)    61(0.61%)    83(0.83%)    55(0.55%)    46(0.46%)    29(0.29%)
        NUMBER TESTED (% PASSED)    461(88.29%)  451(86.47%)  397(79.09%)  352(84.38%)  287(83.97%)  200(85.50%)

 17   ROUND DID NOT ENGAGE TARGET     1(0.01%)     1(0.01%)     4(0.04%)     2(0.02%)     0(0.00%)     0(0.00%)
        NUMBER TESTED (% PASSED)    407(99.75%)  390(99.74%)  314(98.73%)  297(99.33%)  241(100.0%)  171(100.0%)

 18   ROUND ENGAGED BUT DID NOT HIT 105(1.05%)    83(0.83%)    74(0.74%)    54(0.54%)    43(0.43%)    49(0.49%)
        NUMBER TESTED (% PASSED)    406(74.14%)  389(78.66%)  310(76.13%)  295(81.69%)  241(82.16%)  171(71.35%)

 19   ROUND HIT BUT DID NOT KILL    156(1.56%)   159(1.59%)   104(1.04%)   128(1.28%)    96(0.96%)    64(0.64%)
        NUMBER TESTED (% PASSED)    301(48.17%)  306(48.04%)  236(55.93%)  241(46.89%)  198(51.52%)  122(47.54%)

            ***** KILLS             145          147          132          113          102           58 *****

        PROB(KILL/DCSN)=            1.45%        1.47%        1.32%        1.13%        1.02%         .58%
        PROB(KILL/ATMP ENG)=       14.40%       14.60%       13.11%       11.22%       10.13%        5.76%
        PROB(KILL/SHOT)=           20.66%       20.94%       18.80%       16.10%       14.53%        8.26%
        PROB(LOS)-                 88.77%       84.54%       80.05%       72.79%       60.77%       47.16%
        PROB(SHOT/DCSN)=     7.02%    PROB(SHOT/ATMP ENG)=    69.71%     PROB(ATMP ENG/DCSN)=    10.07%

DESIGNATOR SURVIVABILITY: CASES IN WHICH DESIGNATOR KILLED -   141(1.41%)     NUMBER TESTED (% NOT KILLED) =  657(78.54%)
```

LISTING 6-4  CASE RESULTS PAGE

(c) the number of times the test was failed,

(d) the percentage of the total occasions that failed the test

(e) The number of potential COPPERHEAD fire missions on which the test was performed, (this number is equal to the total number of potential COPPERHEAD fire missions minus the number of potential missions that failed prior tests).

(f) The percentage of those potential COPPERHEAD fire missions on which the test was performed that passed the test.

Following the seventh abort cause is a line giving the number of attempted engagements. This is the number of replications (out of the total original number of potential COPPERHEAD missions) that passed the first seven tests and have, therefore, reached the point where the D.O. attempts to call for COPPERHEAD fire.

The next four cause-of-abort lines (abort causes numbers 8 through 11) consist of those tests that must be passed before the battery will fire a COPPERHEAD mission. Each of these abort causes has one line giving information as described in (a) through (f) above in this section.

Following the eleventh cause-of-abort line is a line giving number of shots. This is the number of replications for which the mission reached the point where the battery actually fired COPPERHEAD (i.e., the mission passed tests 1 through 11). The number of shots is the number of missions fired, not the number of individual rounds fired.

The final eight abort causes (numbers 12 through 19) are round aborts which have two lines each in the printout.

The first line of each round abort cause gives:

(a) The number of the abort cause (test),

(b) a label describing the cause of abort,

(c) the number of times the test was failed for the first round,

(d) the percentage of the total occasions for which the first round failed the test.

The line continues with items (c) and (d) repeated but for the second and subsequent rounds up to a total of NRF rounds ($NRF \leq 6$).

The second line of each round abort gives

(e) the number of COPPERHEAD mission first rounds on which the test was performed,

(f) the percentage of those COPPERHEAD fire mission first rounds on which the test was performed that passed the test.

The second line continues with items (e) and (f) repeated but for the second and subsequent rounds of the COPPERHEAD fire mission.

Following the final (19th) abort cause line is a line giving kills. This line gives the number of replications for which the first round of the COPPERHEAD mission achieved a kill, the number of replications for which the second round of the COPPERHEAD mission achieved a kill, and so on through the final round of the COPPERHEAD mission.

Next, the printout gives a number of probabilities derived from the case just simulated: (these probabilities are printed as percentages).

The probability of kill given an occasion for the nth round of the fire mission is printed and is calculated by dividing the number of kills for the nth round by the number of occasions.

The probability of kill given an attempted engagement for the nth round equals the number of kills for the nth round divided by the number of attempted engagements.

The probability of kill given a shot for the nth round equals the number of kills for the nth round divided by the number of shots.

The probability of line-of-sight (LOS) for the nth round is the probability of having LOS when the round arrives (test 12) given that LOS existed immediately prior to firing (test 11). (Note: The preceding applies to the "shooting gallery" LOS model; for random occurrence, the probability of having line-of-sight for the nth round is just equal to the success rate of test 12).

The probability of shot given an occasion is the fraction of potential COPPERHEAD fire missions that actually reach the point where rounds are fired.

The probability of shot given an attempted engagement is the number of shots divided by the number of attempted engagements.

The probability of attempted engagements given an occasion is the number of attempted engagements divided by the number of occasions.

The final line of the results page gives information on designator survivability. This line gives the number of replications in which the designator was killed by direct fire from the target and the percentage of the total replications that number represents. The line also gives the number of replications for which the test of designator kill was performed and the percentage of those replications in which the designator was <u>not</u> killed.

<u>Note</u>: The number of kills shown on the results page differs slightly in its meaning depending on whether "shooting gallery" or "random occurrence" line-of-sight model is used. When "shooting gallery" is used, no attempt is made to account for multiple kills of the same target vehicle (i.e., the number of kills on the nth round is calculated under the assumption that the vehicle the round hit had not already been killed by a previous round).

When "random occurrence" is used, the number of kills on the nth round takes into account the number of vehicles surviving to the nth round.

This difference, of course, has no effect in comparing first round kills computed using the two differing LOS models. Furthermore, for round n there will be no effect from this difference in bookkeeping if (1) $n \leq$ number of vehicles in the target, (2) distance between target vehicles/target velocity < time between rounds, and (3) the D.O. is assumed always to designate the front most target vehicle in view. In other cases, the results obtained by the two methods will have some differences attributable to the difference in bookkeeping between the two methods.

CHAPTER 7

# 7. PROGRAM GENERATED MESSAGES AND DEBUGGING PRINTS

The COPE program has been designed so that most of the common errors will cause the program to come to a stop and print an error message. Of course, it is not possible to anticipate every possible error and the user can expect occasionally to encounter a run that terminates with a dump or a system generated error message. However, the current version of COPE has been used for hundreds of runs now and has not terminated in anything other than the normal program termination or one of the anticipated error stops.

The messages generated by COPE can be divided into five classes:

(1) program generated $$$ comments,

(2) program generated $$ comments,

(3) special program print-outs of a warning nature that do not result in program termination

(4) special program prints that occur with abnormal program terminations and which give information useful in locating the cause of termination.

(5) STOP messages that are printed in the day file. (These STOPS include the subroutine or function name in which the stop occurs.)

The messages are now explained. They are ordered alphabetically (by first line) within each class. (Note that some messages include more than one line).

## 7.1 Program Generated $$$ Comments.

YOU HAVE SEQUENCE NUMBERS IN COLUMNS 73-78 OF TAPE 5. THEY WILL BE REMOVED BY THE PROGRAM;
HOWEVER, IF THE TEMPORARY OPTION WAS USED TO READ IN ANY DATA, THE RESULTS MAY BE SUSPECT.

This message is generated in subroutine SEPREC and is part of a modification added to handle a Control Data peculiar problem. Users of COPE frequently create TAPE 5 (the file of case descriptors) using the CDC editor in its default FORTRAN format and then save the resulting file by the command SAVE, LFN (where LFN is the local file name). The problem is that the editor then inserts sequence numbers in columns 73-78 of the saved file. Now one could easily avoid this problem by using SAVE, LFN, N rather than SAVE, LFN (the additional N

causing no sequence numbers to be saved). However, to avoid aborting runs in which sequence numbers are present, COPE was modified to check the first non-comment card (or card image) of file TAPE 5 for sequence numbers: if columns 73-78 are filled with numbers and columns 63-72 and 79-80 are blank, then the computer concludes that sequence numbers are present, issues the above comment, and ignores columns 73-78 on all cards of the current run.

As mentioned in the message itself, this ignoring of columns 73-78 could adversely affect the results if the TEMPORARY option is used, since that option quite likely has important information in columns 73-78. In this event, the file TAPE 5 should be created in the editor with 80 character lines (i.e., use CDC command F,CH=80 before creating the file). Then when it is saved, the sequence numbers will be in columns not read by the program.

For keywords other than TEMPORARY, there is not likely to be any adverse effect since the case descriptor character strings on the cards will usually terminate before column 63 and hence the deletion of columns 73-78 will have no effect.

Finally, if the computer should erroneously conclude that sequence numbers are not present when they really are (as could happen if any of columns 63-72 or 79-80 of the first non-comment card were not blank), then the computer would treat the sequence numbers as part of the keyword or optionword and a different error would bring the program to a halt when that keyword or optionword was not found (see section 7.4.5).

One may wish to remove this feature if using a non-CDC computer.

## 7.2 Program Generated $$ Comments.

### 7.2.1 CHOICE OF option word FOR RESPONSE TIME DATA IS INCONSISTENT WITH TIME OF option word FOR WEATHER DATA.

The first option word is DAY or NIGHT while the second one is 0600, 1400, or 2200.

This message merely warns the user that one of the following pairs of a RESPONSE TIME data option word and a WEATHER data option word is being used:

| RESPONSE TIME option word | WEATHER option word |
|---|---|
| DAY | 2200 |
| NIGHT | 0600 |
| NIGHT | 1400 |

AD-A100 285 ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY ABERDEEN PROV--ETC F/G 19/1
COPPERHEAD OPERATIONAL PERFORMANCE EVALUATION (COPE): COMPUTER --ETC(U)
MAR 81 R S SANDMEYER

UNCLASSIFIED AMSAA-TR-318 NL

2 OF 6

AD A
100285

These are all considered inconsistent pairs since daylight is usually gone by 2200. On the other hand, day light is usually present at 0600 and 1400.

The choice of one of these inconsistent pairs will not abort the program, but merely result in the generating of this message and the executing of the case with the inconsistent data selected.

### 7.2.2 IN CCPLOT FOR CASE nn WITH ICCPFG= 1. NO PLOT PRODUCED.

nn is the case number.

This message merely means that the user wanted a Cal Comp plot for case number nn, but since the plot routine is not yet implemented, no plot was produced.

### 7.2.3 IN PPLOT FOR CASE nn WITH IPPFG = 1. NO PLOT PRODUCED.

nn is the case number.

This message merely means that the user wanted a printer plot for case number nn, but since the plot routine is not yet implemented, no plot was produced.

### 7.2.4 NO ENDF CARD FOUND BUT ONE ASSUMED SINCE END OF INPUT FILE ENCOUNTERED.

This message occurs when no ENDF card has been put at the end of TAPE 5 (INPUT). It merely means that the program reached the end-of-file without encountering an ENDF card and then proceeded as though it had found one there.

### 7.2.5 NO 'FIRST CASE' OR 'NEXT CASE' CARD FOUND --- NEXT CASE, RD OPTION ASSUMED.

This message means that inputs for a new case (possibly the first case) have been encountered, but were not preceded by a 'FIRST CASE' or 'NEXT CASE' card. In this event, the program proceeds as though a NEXT CASE, RD card had been encountered prior to the new case's inputs.

### 7.2.6 PEDATA FOR: TR=xxx. DT= n VEL= yy. GTR= zzz. REFL= a.aa ANGLET= bbb. DEFT= ccc. TGTHD= dd. SKSEN= ee.

xxx., n, yy., zzz., a.aa, bbb., ccc., dd., and ee. are respectively the response time, designator type number, target velocity, gun-to-target range, reflectivity of target, angle T, deflection bias, target heading, and seeker sensitivity used to generate the PEDATA used in this run. This message occurs only when the OVERRIDE option is used with PEDATA.

The units used with each quantity are those normally used on the corresponding keyword cards.

7.2.7 RECORD NAMED: Record Name USED TO OVERRIDE FOR Data Block Name.

Here record name is the name of the record on the word addressible mass storage file that is selected by the OVERRIDE option and data block name is the name of the data block to be filled by this override.

This print occurs only with each data block in a given case for which the OVERRIDE is in effect.

7.2.8 TEMPORARY OPTION IN EFFECT FOR Data Block Name.

Here data block name is the name of the data block filled by use of the TEMPORARY OPTION.

This print occurs only with each data block in a given case for which the TEMPORARY option is in effect.

7.2.9 YOUR INPUT LINE: Input Line Characters SUPERSEDES ONE OF YOUR PREVIOUS INPUT LINES FOR THIS CASE.

The input line characters are the contents of the card that supersedes a previous line. This message is only a warning to the user that he has included two (or more) cards with the same keyword in a single case. When this happens, only the last such card has an effect on the choice of case descriptors.

If two PLOT cards (section 4.4.5) are included in a case (one for CALCOMP and one for PRINTER), then this message will be produced. In such a case, both plot types are set according to the input cards even though the message says the last one superseded the earlier one; this is the one case in which this message can be safely ignored.

In most cases, one will want to check the input cards and probably re-run the case so that no more than one card is used for each keyword. Note that an OVERRIDE or TEMPORARY card applied to a certain keyword counts as an input card with that keyword for purposes of this message (for example, putting an OVERRIDE, W, JUN1400021 card and a W, J, 0600 card in the same case will produce the message of this section).

## 7.3 Special Program Print-Outs of a Warning Nature That Do Not Result in Program Termination.

7.3.1 IN OUTPUT. PROBLEM 1 J= nnnnn RNDRED=xxxx. NKILL = mmmmm.

nnnnn is the number of the round, xxxx. is the number of kills on the Jth round for this case calculated by adding one each time a kill occurs, and mmmmm is the number of kills on the Jth round for this case calculated by subtracting the number of aborted missions and Jth rounds from the number of replications.

This print occurs only if the Jth round kills (J runs from 1 to NRF) computed by the two methods differ. It indicates a serious problem with the program.

#### 7.3.2 SERIOUS PROBLEM! KTEST AND NTEST ARRAYS DO NOT AGREE II= nnnnn JJ = mmmmm KTEST ARRAY: (Followed By A 20 x 6 Matrix).

nnnnn is the abort number and mmmmm is the round number (or if nnnnn corresponds to a mission abort, mmmmm is 1).

II and JJ are the indices of the first entry in the KTEST and NTEST arrays that do not agree.

The 20 x 6 matrix is the KTEST array where the element in row I column J is KTEST (I, J). The NTEST array elements correspond to the 'number tested' values printed on the case results page where NTEST (I, J) corresponds to the Jth round Ith abort condition (when the abort is a mission abort, J=1).

The KTEST array is filled by incrementing KTEST (I,J) each time the Ith test is made for the Jth round (if the test is for a mission abort J=1). The NTEST array is filled by setting NTEST (I,J) equal to the number of replications minus the number of missions (and Jth rounds) aborted by tests 1 through I-1.

If NTEST and KTEST differ, then the program is not behaving correctly.

### 7.4 Special Program Prints That Occur with Abnormal Program Termination and Which Give Information Useful in Locating the Cause of Termination.

#### 7.4.1 ERROR IN INPUT. LINE: Character String WITH IT1=nnnnn IS IN ERROR.

The character string is the card image read from TAPE 5 (INPUT) that is in error and nnnnn is the keyword number.

The error occurs when a keyword match is found for the input, but no code has been provided to handle the input. This is a serious error and means that a program change is required.

This message corresponds to STOP in section 7.5.15.

#### 7.4.2 IN COMMNT. TOO MANY $$ COMMENTS (>20): (Followed By 20 Numbered Lines of Comments, Then:) THE FOLLOWING COMMENT CAUSED OVERFLOW OF CMENT ARRAY: (Followed by a 21st Comment Line).

The CMENT array is dimensioned to allow up to 20 $$ comments per case. This message means that 21 $$ comments were generated for the current case and consequently the CMENT array overflowed. The $$ comment count includes both program generated and user comments.

The corrective action is either to reduce the number of comments for the case or increase the second dimension of the CMENT array.

Note: One possible cause of too many comments is the omission of one or more END cards.

This message corresponds to the STOP of section 7.5.1.

### 7.4.3 INPUT ATTEMPTED TO READ RECORD NAMED Record Name BUT NO SUCH RECORD WAS FOUND.

The record name is the name of the record which the program attempted to read from the word addressible mass storage file but which could not be found on that file.

This message means the desired record does not exist on the mass storage file. In order to create it, one of the pre-processor programs must be run. If the OVERRIDE feature was used, try preceding the record name with a $ (dollar sign), as described in section 4.5.3.

This message is accompanied by a trace back print of the subroutines called to produce the message. This can be used to locate which call to READMS actually caused the abnormal termination.

This message occurs with the STOP in section 7.5.21.

### 7.4.4 IN FINDIT. KEYWORD: Keyword AND CORRESPONDING VALUE OF IT: Value ARE NOT ALLOWED WITH THE Option Word OPTION.

Here keyword is the keyword character string, value is the keyword number, and option word is an option word character string.

This message occurs when one attempts to use OVERRIDE, TEMPORARY, or RESET features with an option word that is a keyword other than the name of one of the data blocks.

This message corresponds to the STOP of section 7.5.5.

### 7.4.5 IN FINDIT. NO MATCH FOUND FOR LINE IB= nnn OF THIS B ARRAY: (Followed By Up to 10 Lines of Character Strings).

Here nnn is the value of IB.

IB is the number of the character string for which no match was found. If IB=1, there was no keyword match (this corresponds to the stop of section 7.5.4). If IB=2, then one could be in the situation of the STOP of section 7.5.4, section 7.5.6 or section 7.5.7.

This error usually means the keyword or option word found in line IB was misspelled or does not exist.

#### 7.4.6 IN NUMRIC. LINE nnn EXCEEDS ALLOWABLE FIELD LENGTH FOR NUMERIC DATA: (Followed By Up to 10 Lines of Character Strings).

Here nnn is the number of the line in the listing of character strings which contains a numerical string of more than 10 characters (including "-" sign and "." if present).

The corrective action is to change the number to one of less than 10 characters (when including "-" and "."). With the parameters used in COPE there should never be occasion to use a number requiring more than 10 characters.

(Note: If the number is written on input without a decimal point, then the program adds a decimal point after the last digit. Hence, if the decimal point is omitted the number should not exceed nine characters).

This error corresponds to the STOP of section 7.5.23.

#### 7.4.7 IN NUMRIC. LINE nnn HAD TOO MANY DECIMAL POINTS: (Followed By Up To 10 Lines of Character Strings).

Here nnn is the number of the line in the listing of character strings which contains more than one ".".

The corrective action is to rewrite the input line so that no option word has more than one decimal point.

This error corresponds to the STOP of section 7.5.22.

#### 7.4.8 IN SMPLCD WITH RANDOM NUMBER GREATER THAN 1.0 RN=xx.xxxxxxx.

Here xx.xxxxxx is the random number generated for use in SMPLCD.

This error will occur only if the random number generator malfunctions and produces a number 1.0. If this error occurs, a trace of subroutine calls is produced and the STOP of section 7.5.31 occurs.

### 7.5 STOP Messages That are Printed in the Dayfile.

7.5.1 STOP IN COMMNT: TOO MANY $$ COMMENTS. This message occurs when too many $$ comments are encountered in one case. See section 7.4.2.

7.5.2 STOP IN COPE: NORMAL PROGRAM TERMINATION. This is the desirable stop message. It means that the program terminated after processing all the inputs and that no abnormal stopping conditions were encountered. Of course, it does not guarantee that the program did what the user expected, and there may be some warning diagnostic messages (of types 1, 2, or 3 above) produced.

7.5.3 STOP IN DFCHK: ERROR NUMBER 1. This stop occurs when the designator-to-target range cannot be bracketed by the range values in the DFFOKL array. This indicates either an error in the range calculation or a need for larger range values in the DFFOKL array.

7.5.4 STOP IN FINDIT: ERROR NUMBER 1. This message means that a search of the CHAR array has found no keyword to match that of the current input card. (If the OVERRIDE, TEMPORARY, or RESET option is used, then this message will also occur if the second character string on the card cannot be matched from the CHAR array).

This stop corresponds to the print-out of section 7.4.5.

7.5.5 STOP IN FINDIT: ERROR NUMBER 2. This message occurs when one attempts to use the OVERRIDE, TEMPORARY, or RESET option with anything other than one of the eight data block names. (See section 4.5.)

See section 7.4.4.

7.5.6 STOP IN FINDIT: ERROR NUMBER 3. This message occurs when no match for a given option word from an input card can be found among the allowable option words for the input card's keyword.

See section 7.4.5.

7.5.7 STOP IN FINDIT: ERROR NUMBER 4. This message means the same as that in 7.5.6, but occurs at a different point in subroutine FINDIT.

7.5.8 STOP IN FPRCNT: ERROR NUMBER 1. This message means that function FPRCNT was called to format into a percentage a number greater than or equal to 100.05. There should never be any percentages greater than 100 in this program, so if this message is obtained, there are major problems in the bookkeeping processes. In particular, the NABORT array and NTEST array are probably filled with wrong numbers.

7.5.9 STOP IN GETRNG: TARGET VELOCITY = 0.0. This message means a target velocity of zero was input. This is not allowed since division by zero in calculating line-of-sight duration would cause division overflow on the computer.

If one really wishes to use zero velocity, it is suggested that one use a velocity of, say, .001 which will create line-of-sight durations so great that the end result will not differ significantly from those of a stationary target.

A subroutine call trace is also produced.

7.5.10 STOP IN GETVIS: RANDOM NUMBER TROUBLE 1. This message means that the attempt to sample weather data failed under cloud free line-of-sight conditions.

This probably means that there is an error in the PRGCFL array or the random number generator is malfunctioning.

A subroutine call trace is also produced.

7.5.11 STOP IN GETVIS: RANDOM NUMBER TROUBLE 2. This message means that the attempt to sample weather data failed under no cloud free line-of-sight conditions.

This probably implies that there are invalid numbers in the W array or the random number generator is malfunctioning.

A subroutine call trace is also produced.

7.5.12 STOP IN HEADER: ERROR NUMBER 1. This error occurs only if IOVER (J) has a value other than 0, 1, or 2 for a value of J = 21, 22, 23, 24, 25, 26, 27, or 28.

This should never happen. If it does there is a serious problem somewhere in the input or the random access mass storage file.

7.5.13 STOP IN HITCHK: FAILED TO BRACKET RANGE. This stop occurs when the designator-to-target range cannot be bracketed by the range values in the RNGPST array. This indicates either an error in the range calculation or a need for longer range values in the RNGTTF array.

7.5.14 STOP IN INPUT: EOF ENCOUNTERED. This stop occurs when an end-of-file is encountered on TAPE 5 after an END card (instead of after the expected ENDF card).

7.5.15 STOP IN INPUT: ERROR NUMBER 1. This stop corresponds to the message of section 7.4.1.

7.5.16 STOP IN INPUT: EXTRA NEXT CASE CARD. This stop occurs when two (or more) NEXT CASE cards occur within the same case (i.e., without an intervening END card). The program will not accept more than one NEXT CASE card per case. (All cards lying between consecutive END cards are considered to form a case. For this purpose, the beginning and end of TAPE 5 may both be regarded as END cards.)

7.5.17 STOP IN INPUT: FIRST CASE CARD IN SUBSEQUENT CASE. This stop occurs if a FIRST CASE card is encountered anywhere on TAPE 5 other than in the first case. Cases other than the first case should be headed by NEXT CASE cards rather than FIRST CASE cards.

7.5.18 STOP IN INPUT: PARAMETERIZED RESPONSE TIME <=0.0. This stop occurs when attempting to play a parameterized response time less than or equal to zero. Such times are not allowed in the program.

7.5.19 STOP IN LOSCHK: ERROR WITH RANDOM OCCURRENCE OPTION. This message means that the designator-to-target range cannot be bracketed by the range value in the RNGPLS array. This indicates either an error in the range calculation or a need for larger range values in the RNGPLS array.

7.5.20 STOP IN MINTRN: FAILED TO BRACKET RANGE. This stop occurs when the designator-to-target range cannot be bracketed by the range values in the RNGPST array. This indicates either an error in the range calculation or a need for larger range values in the RNGPST array.

7.5.21 STOP IN NOREC: ATTEMPTED TO READ NON EXISTENT RECORD. This stop corresponds to the message of section 7.4.3.

A subroutine call trace is also produced.

7.5.22 STOP IN NUMRIC: ERROR NUMBER 1. This stop corresponds to the error message of section 7.4.7.

7.5.23 STOP IN NUMRIC: ERROR NUMBER 2. This stop corresponds to the error message of section 7.4.6.

7.5.24 STOP IN PECHK: FAILED TO BRACKET DELTA T. This message means that the time delay cannot be bracketed by the DLTT array time values. This indicates either an error in calculating DELTAT or a need for larger time values in the DLTT array.

7.5.25 STOP IN PENAME: ERROR NUMBER 1. This stop means that one of the input values that determines which PEDATA record is to be used had a value not found in the XVALUE array. (See section 5.4.)

7.5.26 STOP IN PENAME: ERROR NUMBER 2. This stop means that an illegal index value was assigned to NP in subroutine PENAME. (See section 5.4.)

7.5.27 STOP IN PENAME: ERROR NUMBER 3. This stop means that a response time less than zero or greater than or equal to 999.5 was input. This is outside the allowable range of response times.

7.5.28 STOP IN PENAME: (PEIDNT) ERROR NUMBER 4. This stop means that the PEDATA record name could not be decoded successfully. (See section 5.4.)

7.5.29 STOP IN SEPREC: ERROR NUMBER 1. This error occurs only if the number of keywords and/or option words on an input card is greater than 10 or less than or equal to zero.

7.5.30 STOP IN SMPLCD: FAILED TO BRACKET RANDOM NUMBER. This stop occurs when the random number obtained from the uniform random number (0 to 1) generator cannot be bracketed by the cumulative probability values describing the distribution to be sampled. It indicates either a malfunctioning random number generator or an incomplete description of the distribution to be sampled.

A subroutine call trace is also produced.

7.5.31 STOP IN SMPLCD: RANDOM NUMBER TOO LARGE. This stop corresponds to the error message of section 7.4.8.

A subroutine call trace is also produced.

## 7.6 Echo Printing.

The COPE program uses TAPE 8 (i.e., logical output unit 8) to echo inputs, to print the values of the case parameters, and to print the initial random number seeds. TAPE 8 is normally a scratch file that is lost when the job is completed; however, if one wishes to see the contents of TAPE 8 for debugging or verification purposes, then one need merely list it or catalog it for future reference.

A sample of TAPE 8 contents is shown in Listings 7-1 and 7-2.

For each case that is run there are two sections of output on TAPE 8. The first section is a listing of the input options (i.e., the case descriptor cards on TAPE 5) that apply to the case.

The second section (Data Check) displays the values of most of the data that describe the case. This section includes data from the word addressable mass storage file (or its equivalent obtained by using the TEMPORARY option), data obtained directly from INPUT cards, and a few computed values. Together, these data give the complete numerical description of the case to be simulated. (This echo print data check has two forms, a long form and a short form. The long form gives all of the data describing the case; the short form gives it all except the PEDATA (i.e., the INDEX and PETBL arrays). (See section 4.4.6.)

Listing 7-1 gives an example of what is printed on TAPE 8 when the short form of the echo print is used. Listing 7-2 shows the TAPE 8 printout when the echo print long form is used.

The values listed in the Data Check section of the TAPE 8 printout are not identified by any printout labeling. Instead, the user must locate the value of a particular variable on the printout by matching the printout lines with the WRITE statements in subroutine ECHO. If the user intends to use this Data Check section frequently, then perhaps some labeling should be added.

In addition to the two sections of output for each case, TAPE 8 also includes the random number seeds. These are printed immediately following the first case data check. Since the random number seeds used for each case are the same, the print of random number seeds is not repeated following the data check prints for subsequent cases.

Note that the column numbers at the top and bottom of each page of Listings 7-1 and 7-2 are not part of the output produced on TAPE 8. They are included here only for the reader's convenience.

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

************************************************** INPUT OPTIONS FOR CASE    1 *****************************************
FIRST CASE
$$$ THIS COMMENT WILL APPEAR ON THE $$$ COMMENT PAGE
WEATHER,JUNE,0600
ACORNCDIST,CLOSE
PLOT,PRINTER,ON
END
 -   -   -   -   -   -   -   -   -   -   -   -   -  DATA CHECK FOR CASE    1  -   -   -   -   -   -   -   -   -   -   -
10000     6
     .0100     .7506     .3700     .0600
  484   846
 7000.0000
   31.0000   20.0000
    3.0000        10   61.0000         1
     .5590     .2200     .6900
     .0070     .0040    0.0000     .0040     .0220    0.0000     .0180     .0040     .0510     .0320     .3430
     .0110    0.0000    0.0000     .0040     .0040     .0040     .0110     .0140    0.0000     .0220     .4480
     .0180    0.0000    0.0000    0.0000    0.0000     .0150
     .0070    0.0000    0.0000    0.0000    0.0000     .0040
     .0040    0.0000    0.0000    0.0000    0.0000     .0110
    0.0000     .0070    0.0000    0.0000    0.0000     .0110
     .0040     .0040    0.0000    0.0000    0.0000     .0360
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0040    0.0000    0.0000    0.0000    0.0000     .0330
    0.0000    0.0000    0.0000    0.0000     .0040     .0040
    0.0000    0.0000     .0040    0.0000    0.0000     .0510
    0.0000    0.0000     .0040    0.0000    0.0000     .0360
     .0040     .0070     .0110     .0150     .0110     .2520
     .0040     .0070     .0040     .0040     .0040     .0150
    0.0000    0.0000    0.0000    0.0000    0.0000     .0070
     .0070    0.0000     .0070    0.0000    0.0000     .0070
     .0070    0.0000    0.0000    0.0000    0.0000    0.0000
     .0040    0.0000    0.0000    0.0000    0.0000     .0070
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000     .0040    0.0000    0.0000     .0220
     .0070    0.0000     .0040    0.0000    0.0000     .0110
    0.0000    0.0000     .0040     .0040    0.0000     .0220
    0.0000     .0040    0.0000    0.0000    0.0000     .0220
     .0110     .0070     .0040     .0040     .0180     .2120
    6   11
     .4850     .5180     .5610     .4440
    0.0000    0.0000    0.0000     .2700     .1200     .6100
    1.0000     .8800    1.0000
    0.0000     .1200    0.0000
    0.0000    0.0000    0.0000
    0.0000    1.0000
   11   11     3
    0.0000     .1000     .2000     .3000     .4000     .5000     .6000     .7000     .8000     .9000    1.0000
    0.0000  400.0000  650.0000  990.0000 1300.0000 1600.0000 2000.0000 2500.0000 3000.0000 3700.0000 7000.0000
    0.0000     .1000     .2000     .3000     .4000     .5000     .6000     .7000     .8000     .9000    1.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-1 SHORT FORM OF ECHO PRINT

(PAGE 1 OF 3)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

   25.0000   30.0000   40.0000   50.0000   70.0000  120.0000  140.0000  180.0000  280.0000  500.0000 1100.0000
   25.0000   30.0000   40.0000   50.0000   70.0000  120.0000  150.0000  200.0000  420.0000  900.0000 1520.0000
   75.0000   40.0000   60.0000  100.0000  140.0000  190.0000  210.0000  240.0000  310.0000  590.0000 1600.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
 2000.0000 2500.0000 9999.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    3.0000    2.0000
    2.0000    2.0000
    1    1    1    2    0    0
 1000.0000
    0.0000  500.0000 1000.0000 1500.0000 2000.0000 3000.0000 3001.0000 9999.0000    0.0000    0.0000
    0.0000     .5000     .4000     .3000     .2000     .1000    0.0000    0.0000    0.0000    0.0000
    1.0000     .3000     .2000     .0500     .0100    0.0000    0.0000    0.0000    0.0000    0.0000
   25.0000    8.0000     .1000    0.0000    0.0000   60.0000
 NAME OF PFDATA RECORD USED: PEODAJ:AJS
  16
    0.0000     .9900     .9500     .9900     .8000     .9000     .6000
  500.0000     .9500     .9000     .9500     .7500     .8500     .5500
 1000.0000     .9500     .8500     .9000     .7000     .8000     .4000
 1500.0000     .9000     .8000     .8500     .6000     .6000     .3000
 2000.0000     .8500     .8000     .8500     .6500     .7500     .3000
 2500.0000     .8000     .7500     .7000     .6000     .7000     .1500
 3000.0000     .7500     .7000     .6500     .5000     .6000     .0500
 3500.0000     .7000     .6500     .6000     .4500    0.0000    0.0000
 4000.0000     .6500     .6000     .5500     .4000    0.0000    0.0000
 4500.0000     .6000     .5500     .5000     .3500    0.0000    0.0000
 5000.0000     .5500     .5000     .4500     .3000    0.0000    0.0000
 5500.0000     .5000     .4500     .4000     .2500    0.0000    0.0000
 6000.0000     .4500     .4000     .3500     .2000    0.0000    0.0000
 6500.0000     .4000     .3500     .3000     .1500    0.0000    0.0000
 7000.0000     .3500     .3000     .2500     .1000    0.0000    0.0000
 9000.0000     .3000     .2000     .1000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000   30.0000   90.0000  150.0000  300.0000 9000.0000
     .5000     .7000     .5000     .8000     .7000     .9500     .6000    0.0000    0.0000    0.0000
     .5000     .6000     .4500     .7500     .7000    1.0000     .3000    0.0000    0.0000    0.0000
    8
    0.0000  500.0000 1000.0000 1500.0000 2000.0000 2500.0000 3000.0000 9000.0000    0.0000    0.0000
     .0700     .0700     .1300     .1500     .1600     .1700     .1800     .1800    0.0000    0.0000
     .9000     .9000     .8200     .7800     .7600     .7400     .7300     .7300    0.0000    0.0000
     .0300     .0300     .0500     .0700     .0800     .0900     .0900     .0900    0.0000    0.0000
   20.0000   20.0000   20.0000    0.0000    0.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-1 SHORT FORM OF ECHO PRINT - CONT'D

(PAGE 2 OF 3)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
   69.1900   69.1900   69.1900   69.1900   46.3200   46.3200   46.3200   46.3200
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    9
    0.0000     .1250     .2500     .3750     .5000     .6250     .7500     .8750    1.0000    0.0000
    0.0000    5.0000   10.0000   18.0000   24.0000   36.0000   52.0000   87.0000  174.0000    0.0000
   75.0000   75.0000   75.0000   75.0000   75.0000   75.0000
    0.0000  106.0000     .9150     .9750     .9750     .9750    0.0000
    1
*********************************************** END OF DATA CHECK FOR CASE   1 ****************************************
RANDOM NUMBER SEEDS
 18512791  13884593  43967877  16198691  62480667  30083283  39339679  46281975  34711481   9256395
  6942297  55538371  41653777  64794765  48596073  53224271  23140987  50910173   4628197  37025581
 27769185  20826889  32397383  57852469  60166567  11570493  25455087   2314099

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-1 SHORT FORM OF ECHO PRINT - CONT'D

(PAGE 3 OF 3)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

************************************************ INPUT OPTIONS FOR CASE    2 ******************************************
NEXT CASE,DR
DESIGTYPE,MULE
RESPTIME,PARAM,150,N
CONTROL,ECHO,LONG
PLOT,PRINTER,OFF
$ THE OVERRIDE CARD WHICH FOLLOWS OVERRIDES THE NORMAL CHOICE OF
$ PE DATA WITH THE PE DATA RECORD NAMED PEOOAJAAJS
$  NOTE THE USE OF THE $ BEFORE THE RECORD NAME
OVERRIDE,PEDATA,$PEOOAJAAJS
END
 -   -   -   -   -   -   -   -   -   -   -   -   - DATA CHECK FOR CASE    2 -   -   -   -   -   -   -   -   -   -   -   -
10000     6
     .0100     .7506     .3700     .9600
  484  846
 2600.0000
   31.0000   20.0000
    2.0000        10   61.0000         1
     .5590     .2200     .6900
     .0070     .0040    0.0000     .0040     .0220    0.0000     .0180     .0040     .0510     .0320     .3430
     .0110    0.0000    0.0000     .0040     .0040     .0040     .0110     .0140    0.0000     .0220     .4480
     .0180    0.0000    0.0000    0.0000    0.0000     .0150
     .0070    0.0000    0.0000    0.0000    0.0000     .0040
     .0040    0.0000    0.0000    0.0000    0.0000     .0110
    0.0000     .0070    0.0000    0.0000    0.0000     .0110
     .0040     .0040    0.0000    0.0000    0.0000     .0360
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0040    0.0000    0.0000    0.0000    0.0000     .0330
    0.0000    0.0000    0.0000    0.0000     .0040     .0040
    0.0000    0.0000     .0040    0.0000    0.0000     .0510
    0.0000    0.0000     .0040    0.0000    0.0000     .0360
     .0040     .0070     .0110     .0150     .0110     .2520
     .0040     .0070     .0040     .0040     .0040     .0150
    0.0000    0.0000    0.0000    0.0000    0.0000     .0070
     .0070    0.0000     .0070    0.0000    0.0000     .0070
     .0070    0.0000    0.0000    0.0000    0.0000    0.0000
     .0040    0.0000    0.0000    0.0000    0.0000     .0070
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000     .0040    0.0000    0.0000     .0220
     .0070    0.0000     .0040    0.0000    0.0000     .0110
    0.0000    0.0000     .0040     .0040    0.0000     .0220
    0.0000     .0040    0.0000    0.0000    0.0000     .0220
     .0110     .0070     .0040     .0040     .0180     .2120
    6   11
     .4850     .5180     .5610     .4440
    0.0000    0.0000    0.0000     .2700     .1200     .6100
    1.0000     .8800    1.0000
    0.0000     .1200    0.0000
    0.0000    0.0000    0.0000
    0.0000    1.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS

(PAGE 1 OF 17)

```
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000011111111111111111111 COLUMN NOS.
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888899999999990000000000111111111112
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890

   11    11     3
    0.0000     .1000     .2000     .3000     .4000     .5000     .6000     .7000     .8000     .9000    1.0000
    0.0000  400.0000  650.0000  900.0000 1300.0000 1600.0000 2000.0000 2500.0000 3000.0000 3700.0000 7000.0000
    0.0000     .1000     .2000     .3000     .4000     .5000     .6000     .7000     .8000     .9000    1.0000
   25.0000   30.0000   40.0000   50.0000   70.0000  120.0000  140.0000  180.0000  280.0000  500.0000 1100.0000
   25.0000   30.0000   40.0000   50.0000   70.0000  120.0000  150.0000  200.0000  420.0000  900.0000 1520.0000
   25.0000   40.0000   60.0000  100.0000  140.0000  190.0000  210.0000  240.0000  310.0000  590.0000 1600.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
 2000.0000 2500.0000 9999.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    3.0000    2.0000
    2.0000    2.0000
    2    1    1    2    0    0
 1000.0000
    0.0000  500.0000 1000.0000 1500.0000 2000.0000 3000.0000 3001.0000 9999.0000    0.0000    0.0000
    0.0000     .5000     .4000     .3000     .2000     .1000    0.0000    0.0000    0.0000    0.0000
    1.0000     .3000     .2000     .0500     .0100    0.0000    0.0000    0.0000    0.0000    0.0000
   25.0000    8.0000     .1000    0.0000    0.0000   60.0000
 NAME OF PEDATA RECORD USED: PEDOAIAAJS
INDEX ARRAY FOR I=     1
   1    2
   3    4
   5    6
   7    8
   9   10
INDEX ARRAY FOR I=     2
  11   12
  13   14
  15   16
  17   18
  19   20
INDEX ARRAY FOR I=     3
  21   22
  23   24
  25   26
  27   28
  29   30
INDEX ARRAY FOR I=     4
  31   32
  33   34
  35   36
  37   38
  39   40
INDEX ARRAY FOR I=     5
  41   42

0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000011111111111111111111 COLUMN NOS.
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888899999999990000000000111111111112
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 2 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

  43   44
  45   46
  47   48
  49   50
INDEX ARRAY FOR I=     6
  51   52
  53   54
  55   56
  57   58
  59   60
PE ARRAY FOR I=     1
     .2900    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    1.0000     .9200     .5000    0.0000    0.0000    0.0000    0.0000
    1.0000    1.0000    1.0000     .8900     .6600     .1900    0.0000
    1.0000    1.0000    1.0000     .9900     .9800     .9200     .7000
    1.0000    1.0000     .9900     .9900    1.0000     .9900    1.0000
     .9900     .9900     .9900    1.0000     .9900    1.0000    1.0000
     .9900     .9900    1.0000     .9900    1.0000    1.0000     .9900
    1.0000     .9800    1.0000    1.0000     .9900    1.0000    1.0000
     .9900    1.0000    1.0000     .9900     .9900    1.0000    1.0000
    1.0000    1.0000    1.0000    1.0000    1.0000     .9900    1.0000
PE ARRAY FOR I=     2
     .3200    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .9900     .9700     .4700     .0100    0.0000    0.0000    0.0000
    1.0000    1.0000    1.0000     .9800     .6900     .2400    0.0000
     .9900    1.0000    1.0000    1.0000    1.0000     .9700     .8000
    1.0000    1.0000    1.0000    1.0000    1.0000     .9700    1.0000
    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
    1.0000    1.0000     .9900    1.0000    1.0000     .9900     .9800
    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
     .9900    1.0000    1.0000    1.0000    1.0000     .9800    1.0000
    1.0000    1.0000    1.0000    1.0000     .9900    1.0000     .9900
PE ARRAY FOR I=     3
     .4000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .9700     .0100     .4700     .0100    0.0000    0.0000    0.0000
     .9700     .9600     .9500     .8800     .5700     .2800    0.0000
     .9600     .9500     .9700     .9800     .9600     .8800     .6200
     .9800     .9900     .9500     .9500     .9400     .9800     .9700
     .9700     .9500     .9600     .9900     .9600     .9700     .9600
     .9500     .9700     .9300     .9900     .9500     .9600     .9800
     .9600     .9900     .9500     .9500     .9500     .9400     .9600
     .9700     .9300     .9800     .9700     .9500     .9200     .9400
     .9800     .9600     .9800     .9600     .9700     .9600     .9700
PE ARRAY FOR I=     4
     .3400    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .9600     .8600     .4000     .0100    0.0000    0.0000    0.0000
     .9600     .9500     .9400     .8800     .5700     .2500    0.0000
     .9400     .9600     .9800     .9200     .9500     .8400     .6800
     .9900     .9700     .9700     .9600     .9800     .9900     .9600
     .9800     .9500     .9300     .9800     .9500     .9500     .9600

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 3 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

      .9900     .9400     .9500     .9400     .9600     .9700     .9500
      .9400     .9800     .9700     .9500     .9800     .9800     .9700
      .9900     .9400     .9700     .9500     .9700     .9600     .9600
      .9700     .9600     .9700     .9700     .9500     .9500     .9800
PF ARRAY FOR I=    5
      .1900    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .6900     .5600     .2600    0.0000    0.0000    0.0000    0.0000
      .6800     .6500     .7500     .5700     .2900     .1600    0.0000
      .7400     .7500     .7300     .7100     .6900     .5400     .2800
      .6800     .6600     .7600     .8000     .7200     .6700     .7100
      .6900     .7800     .6800     .7400     .7100     .7300     .7600
      .7000     .7300     .6500     .7100     .6600     .7100     .7800
      .7100     .6800     .7100     .7500     .7700     .7500     .7000
      .7000     .7400     .6600     .7300     .6700     .7300     .7500
      .7200     .6800     .6400     .7300     .7100     .7000     .7500
PE ARRAY FOR I=    6
      .1800    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .6800     .5200     .2600    0.0000    0.0000    0.0000    0.0000
      .7100     .7800     .7800     .5300     .2600     .1100    0.0000
      .7000     .8000     .7200     .7400     .7200     .5700     .2200
      .6600     .7200     .7400     .7100     .7300     .7200     .6500
      .7100     .6800     .7300     .6900     .7000     .7800     .7100
      .7500     .7500     .6800     .7700     .6900     .6600     .7400
      .6700     .6500     .6900     .6700     .7000     .7600     .6500
      .7700     .8100     .6600     .7400     .6600     .7100     .7000
      .7300     .7100     .6900     .7100     .6900     .6500     .7300
PF ARRAY FOR I=    7
      .0600    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .3800     .2100     .0700     .0100    0.0000    0.0000    0.0000
      .3600     .3900     .3900     .2300     .0700     .0200    0.0000
      .4000     .3800     .3500     .4000     .3600     .3100     .1100
      .4200     .4300     .4100     .4500     .4200     .4100     .3000
      .4000     .4100     .3100     .4100     .3200     .3100     .4500
      .3400     .3800     .4200     .3700     .3400     .2900     .3200
      .4200     .3700     .3800     .2800     .3700     .4300     .4000
      .3600     .3100     .4600     .3800     .3600     .3800     .2800
      .3500     .3100     .4100     .3500     .4600     .3500     .2700
PF ARRAY FOR I=    8
      .1000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .3700     .2800     .0700    0.0000    0.0000    0.0000    0.0000
      .4700     .4200     .3300     .1600     .1100     .0200    0.0000
      .3300     .3800     .3400     .3500     .3700     .2300     .0800
      .3700     .4700     .3500     .3100     .3600     .3600     .3700
      .3300     .4500     .4200     .3100     .3300     .3800     .2600
      .3100     .4100     .3200     .3400     .4100     .3400     .3600
      .3600     .3800     .2600     .3400     .3500     .4600     .3900
      .3600     .3100     .3000     .3700     .3800     .3600     .3700
      .3700     .3500     .4100     .3700     .2800     .4000     .3900
PF ARRAY FOR I=    9
      .0100    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 4 OF 17)

```
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

      .0500      .0400      .0100    0.0000    0.0000    0.0000    0.0000
      .0400      .0500      .0300     .0100    0.0000    0.0000    0.0000
      .0100      .1100      .0700     .0600     .0400     .0200    0.0000
      .0200      .0600      .0300     .0500     .0500     .0500     .0500
      .0100      .0800      .0200     .0400     .0600     .0500     .0400
      .0600      .0700      .0400     .0800     .0300     .0500     .0800
      .0500      .0200      .0300     .0400     .0300     .0500     .0600
      .0600      .0500      .0300     .0500     .0300     .0500     .0600
      .0500      .0100      .0700     .0200     .0400     .0800     .0500
PE ARRAY FOR I=   10
     0.0000     0.0000     0.0000    0.0000    0.0000    0.0000    0.0000
      .0600      .0300      .0100    0.0000    0.0000    0.0000    0.0000
      .0800      .0300      .0700     .0400    0.0000     .0100    0.0000
      .1100      .0300      .0200     .0300     .0600     .0500     .0200
      .0300      .0600      .0700     .0300     .0400     .0400     .0200
      .0800      .0500      .0200     .0400     .0600     .0400     .0400
      .0500      .0600      .0800     .0200     .0400     .0400     .0400
      .0100      .0700      .0300     .0200     .0500     .0500     .0500
      .0400      .0100      .0700     .0800     .0200     .0100     .0400
      .0300      .0200      .0600     .0300     .0200     .0400     .0400
PE ARRAY FOR I=   11
      .2900     0.0000     0.0000    0.0000    0.0000    0.0000    0.0000
     1.0000      .9200      .5000    0.0000    0.0000    0.0000    0.0000
     1.0000     1.0000     1.0000     .8900     .6600     .1900    0.0000
     1.0000     1.0000     1.0000     .9900     .9800     .9200     .7000
     1.0000     1.0000      .9900     .9900    1.0000     .9900    1.0000
      .9900      .9900      .9900    1.0000     .9900    1.0000    1.0000
      .9900      .9900     1.0000     .9900    1.0000    1.0000     .9900
     1.0000      .9800     1.0000    1.0000     .9900    1.0000    1.0000
      .9900     1.0000     1.0000     .9900     .9900    1.0000    1.0000
     1.0000     1.0000     1.0000    1.0000    1.0000     .9900    1.0000
PE ARRAY FOR I=   12
      .3200     0.0000     0.0000    0.0000    0.0000    0.0000    0.0000
      .9900      .9700      .4700     .0100    0.0000    0.0000    0.0000
     1.0000     1.0000     1.0000     .9800     .6900     .2400    0.0000
      .9900     1.0000     1.0000    1.0000    1.0000     .9700     .8000
     1.0000     1.0000     1.0000    1.0000    1.0000     .9700    1.0000
     1.0000     1.0000     1.0000    1.0000    1.0000    1.0000    1.0000
     1.0000     1.0000      .9900    1.0000    1.0000     .9900     .9800
     1.0000     1.0000     1.0000    1.0000    1.0000    1.0000    1.0000
      .9900     1.0000     1.0000    1.0000    1.0000     .9800    1.0000
     1.0000     1.0000     1.0000    1.0000     .9900    1.0000     .9900
PE ARRAY FOR I=   13
      .4000     0.0000     0.0000    0.0000    0.0000    0.0000    0.0000
      .9700      .9100      .4700     .0100    0.0000    0.0000    0.0000
      .9700      .9600      .9500     .8800     .5700     .2800    0.0000
      .9600      .9500      .9700     .9800     .9600     .8800     .6200
      .9800      .9900      .9500     .9500     .9400     .9800     .9700
      .9700      .9500      .9600     .9900     .9600     .9700     .9600
      .9500      .9700      .9300     .9900     .9500     .9600     .9800

0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 5 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

     .9600    .9900    .9500    .9500    .9500    .9400    .9600
     .9700    .9300    .9800    .9700    .9500    .9200    .9400
     .9800    .9600    .9800    .9600    .9700    .9600    .9700
PF ARRAY FOR I=   14
     .3400   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
     .9600    .8600    .4000    .0100   0.0000   0.0000   0.0000
     .9600    .9500    .9400    .8800    .5700    .2500   0.0000
     .9400    .9600    .9800    .9200    .9500    .8400    .6800
     .9900    .9700    .9700    .9600    .9800    .9900    .9600
     .9800    .9500    .9300    .9800    .9500    .9500    .9600
     .9900    .9400    .9500    .9400    .9600    .9700    .9500
     .9400    .9800    .9700    .9500    .9800    .9800    .9700
     .9900    .9400    .9700    .9500    .9700    .9600    .9600
     .9700    .9600    .9700    .9700    .9500    .9500    .9800
PF ARRAY FOR I=   15
     .1900   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
     .6900    .5600    .2600   0.0000   0.0000   0.0000   0.0000
     .6800    .6500    .7500    .5700    .2900    .1600   0.0000
     .7400    .7500    .7300    .7100    .6900    .5400    .2800
     .6800    .6600    .7600    .8000    .7200    .6700    .7100
     .6900    .7800    .6800    .7400    .7100    .7300    .7600
     .7000    .7300    .6500    .7100    .6600    .7100    .7800
     .7100    .6800    .7100    .7500    .7700    .7500    .7000
     .7000    .7400    .6600    .7300    .6700    .7300    .7500
     .7200    .6800    .6400    .7300    .7100    .7000    .7500
PF ARRAY FOR I=   16
     .1800   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
     .6800    .5200    .2600   0.0000   0.0000   0.0000   0.0000
     .7100    .7800    .7800    .5300    .2600    .1100   0.0000
     .7000    .8000    .7200    .7400    .7200    .5700    .2200
     .6600    .7200    .7400    .7100    .7300    .7200    .6500
     .7100    .6800    .7300    .6900    .7000    .7800    .7100
     .7500    .7500    .6800    .7700    .6900    .6600    .7400
     .6700    .6500    .6900    .6700    .7000    .7600    .6500
     .7700    .8100    .6600    .7400    .6600    .7100    .7000
     .7300    .7100    .6900    .7100    .6900    .6500    .7300
PF ARRAY FOR I=   17
     .0600   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
     .3800    .2100    .0700    .0100   0.0000   0.0000   0.0000
     .3600    .3900    .3900    .2300    .0700    .0200   0.0000
     .4000    .3800    .3500    .4000    .3600    .3100    .1100
     .4200    .4300    .4100    .4500    .4200    .4100    .3000
     .4000    .4100    .3100    .4100    .3200    .3100    .4500
     .3400    .3800    .4200    .3700    .3400    .2900    .3200
     .4200    .3700    .3800    .2800    .3700    .4300    .4000
     .3600    .3100    .4600    .3800    .3600    .3800    .2800
     .3500    .3100    .4100    .3500    .4600    .3500    .2700
PF ARRAY FOR I=   18
     .1000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
     .3700    .2800    .0700   0.0000   0.0000   0.0000   0.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 6 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

      .4700     .4200     .3300     .1600     .1100     .0200    0.0000
      .3300     .3800     .3400     .3500     .3700     .2300     .0800
      .3700     .4700     .3500     .3100     .3600     .3600     .3700
      .3300     .4500     .4200     .3100     .3300     .3800     .2600
      .3100     .4100     .3200     .3400     .4100     .3400     .3600
      .3600     .3800     .2600     .3400     .3500     .4600     .3900
      .3600     .3100     .3000     .3700     .3800     .3600     .3700
      .3700     .3500     .4100     .3700     .2800     .4000     .3900
PE ARRAY FOR I=   19
      .0100    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .0500     .0400     .0100    0.0000    0.0000    0.0000    0.0000
      .0400     .0500     .0300     .0100    0.0000    0.0000    0.0000
      .0100     .1100     .0700     .0600     .0400     .0200    0.0000
      .0200     .0600     .0300     .0500     .0500     .0500     .0500
      .0100     .0800     .0200     .0400     .0600     .0500     .0400
      .0600     .0700     .0400     .0800     .0300     .0500     .0800
      .0500     .0200     .0300     .0400     .0300     .0500     .0600
      .0600     .0500     .0300     .0500     .0300     .0500     .0600
      .0500     .0100     .0700     .0200     .0400     .0800     .0500
PE ARRAY FOR I=   20
     0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .0600     .0300     .0100    0.0000    0.0000    0.0000    0.0000
      .0800     .0300     .0700     .0400    0.0000     .0100    0.0000
      .1100     .0300     .0200     .0300     .0600     .0500     .0200
      .0300     .0600     .0700     .0300     .0400     .0400     .0200
      .0800     .0500     .0200     .0400     .0600     .0400     .0400
      .0500     .0600     .0800     .0200     .0400     .0400     .0400
      .0100     .0700     .0300     .0200     .0500     .0500     .0500
      .0400     .0100     .0700     .0800     .0200     .0100     .0400
      .0300     .0200     .0600     .0300     .0200     .0400     .0400
PE ARRAY FOR I=   21
      .2900    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     1.0000     .0200     .5000    0.0000    0.0000    0.0000    0.0000
     1.0000    1.0000    1.0000     .8900     .6600     .1900    0.0000
     1.0000    1.0000    1.0000     .9900     .9800     .9200     .7000
     1.0000    1.0000     .9900     .9900    1.0000     .9900    1.0000
      .9900     .9900     .9900    1.0000     .9900    1.0000    1.0000
      .9900     .9900    1.0000     .9900    1.0000    1.0000     .9900
     1.0000     .9800    1.0000    1.0000     .9900    1.0000    1.0000
      .9900    1.0000    1.0000     .9900     .9900    1.0000    1.0000
     1.0000    1.0000    1.0000    1.0000    1.0000     .9900    1.0000
PE ARRAY FOR I=   22
      .3200    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .9900     .0700     .4700     .0100    0.0000    0.0000    0.0000
     1.0000    1.0000    1.0000     .9800     .6900     .2400    0.0000
      .9900    1.0000    1.0000    1.0000    1.0000     .9700     .8000
     1.0000    1.0000    1.0000    1.0000    1.0000     .9700    1.0000
     1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
     1.0000    1.0000     .9900    1.0000    1.0000     .9900     .9800
     1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 7 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

     .9900    1.0000    1.0000    1.0000    1.0000     .9800    1.0000
    1.0000    1.0000    1.0000    1.0000     .9900    1.0000     .9900
PF ARRAY FOR I=   23
     .4000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .9700     .9100     .4700     .0100    0.0000    0.0000    0.0000
     .9700     .9600     .9500     .8800     .5700     .2800    0.0000
     .9600     .9500     .9700     .9800     .9600     .8800     .6200
     .9800     .9900     .9500     .9500     .9400     .9800     .9700
     .9700     .9500     .9600     .9900     .9600     .9700     .9600
     .9500     .9700     .9300     .9900     .9500     .9600     .9800
     .9600     .9900     .9500     .9500     .9500     .9400     .9600
     .9700     .9300     .9800     .9700     .9500     .9200     .9400
     .9800     .9600     .9800     .9600     .9700     .9600     .9700
PF ARRAY FOR I=   24
     .3400    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .9600     .8600     .4000     .0100    0.0000    0.0000    0.0000
     .9600     .9500     .9400     .8800     .5700     .2500    0.0000
     .9400     .9600     .9800     .9200     .9500     .8400     .6800
     .9900     .9700     .9700     .9600     .9800     .9900     .9600
     .9800     .9500     .9300     .9800     .9500     .9500     .9600
     .9900     .9400     .9500     .9400     .9600     .9700     .9500
     .9400     .9800     .9700     .9500     .9800     .9800     .9700
     .9900     .9400     .9700     .9500     .9700     .9600     .9600
     .9700     .9600     .9700     .9700     .9500     .9500     .9800
PF ARRAY FOR I=   25
     .1900    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .6900     .5600     .2600    0.0000    0.0000    0.0000    0.0000
     .6800     .6500     .7500     .5700     .2900     .1600    0.0000
     .7400     .7500     .7300     .7100     .6900     .5400     .2800
     .6800     .6600     .7600     .8000     .7200     .6700     .7100
     .6900     .7800     .6800     .7400     .7100     .7300     .7600
     .7000     .7300     .6500     .7100     .6600     .7100     .7800
     .7100     .6800     .7100     .7500     .7700     .7500     .7000
     .7000     .7400     .6600     .7300     .6700     .7300     .7500
     .7200     .6800     .6400     .7300     .7100     .7000     .7500
PF ARRAY FOR I=   26
     .1800    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .6800     .5200     .2600    0.0000    0.0000    0.0000    0.0000
     .7100     .7800     .7800     .5300     .2600     .1100    0.0000
     .7000     .8000     .7200     .7400     .7200     .5700     .2200
     .6600     .7200     .7400     .7100     .7300     .7200     .6500
     .7100     .6800     .7300     .6900     .7000     .7800     .7100
     .7500     .7500     .6800     .7700     .6900     .6600     .7400
     .6700     .6500     .6900     .6700     .7000     .7600     .6500
     .7700     .8100     .6600     .7400     .6600     .7100     .7000
     .7300     .7100     .6900     .7100     .6900     .6500     .7300
PF ARRAY FOR I=   27
     .0600    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .3800     .2100     .0700     .0100    0.0000    0.0000    0.0000
     .3600     .3900     .3900     .2300     .0700     .0200    0.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 8 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

      .4000     .3800     .3500     .4000     .3600     .3100     .1100
      .4200     .4300     .4100     .4500     .4200     .4100     .3000
      .4000     .4100     .3100     .4100     .3200     .3100     .4500
      .3400     .3800     .4200     .3700     .3400     .2900     .3200
      .4200     .3700     .3800     .2800     .3700     .4300     .4000
      .3600     .3100     .4600     .3800     .3600     .3800     .2800
      .3500     .3100     .4100     .3500     .4600     .3500     .2700
PF ARRAY FOR I=   28
      .1000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .3700     .2800     .0700    0.0000    0.0000    0.0000    0.0000
      .4700     .4200     .3300     .1600     .1100     .0200    0.0000
      .3300     .3800     .3400     .3500     .3700     .2300     .0800
      .3700     .4700     .3500     .3100     .3600     .3600     .3200
      .3300     .4500     .4200     .3100     .3300     .3800     .2600
      .3100     .4100     .3200     .3400     .4100     .3400     .3600
      .3600     .3800     .2600     .3400     .3500     .4600     .3900
      .3600     .3100     .3000     .3700     .3800     .3600     .3700
      .3700     .3500     .4100     .3700     .2800     .4000     .3900
PF ARRAY FOR I=   29
      .0100    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .0500     .0400     .0100    0.0000    0.0000    0.0000    0.0000
      .0400     .0500     .0300     .0100    0.0000    0.0000    0.0000
      .0100     .1100     .0700     .0600     .0400     .0200    0.0000
      .0200     .0600     .0300     .0500     .0500     .0500     .0500
      .0100     .0800     .0200     .0400     .0600     .0500     .0400
      .0600     .0700     .0400     .0800     .0300     .0500     .0800
      .0500     .0200     .0300     .0400     .0300     .0500     .0600
      .0500     .0500     .0300     .0500     .0300     .0500     .0600
      .0500     .0100     .0700     .0200     .0400     .0800     .0500
PF ARRAY FOR I=   30
     0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .0600     .0300     .0100    0.0000    0.0000    0.0000    0.0000
      .0800     .0300     .0700     .0400    0.0000     .0100    0.0000
      .1100     .0300     .0200     .0300     .0600     .0500     .0200
      .0300     .0600     .0700     .0300     .0400     .0400     .0200
      .0800     .0500     .0200     .0400     .0600     .0400     .0400
      .0500     .0600     .0800     .0200     .0400     .0400     .0400
      .0100     .0700     .0300     .0200     .0500     .0500     .0500
      .0400     .0100     .0700     .0800     .0200     .0100     .0400
      .0300     .0200     .0600     .0300     .0200     .0400     .0400
PF ARRAY FOR I=   31
      .2900    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     1.0000     .0200     .5000    0.0000    0.0000    0.0000    0.0000
     1.0000    1.0000    1.0000     .8900     .6600     .1900    0.0000
     1.0000    1.0000    1.0000     .9900     .9800     .9200     .7000
     1.0000    1.0000     .9900     .9900    1.0000     .9900    1.0000
      .9900     .9900     .9900    1.0000     .9900    1.0000    1.0000
      .9900     .9900    1.0000     .9900    1.0000    1.0000     .9900
     1.0000     .9800    1.0000    1.0000     .9900    1.0000    1.0000
      .9900    1.0000    1.0000     .9900     .9900    1.0000    1.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 9 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

   1.0000    1.0000    1.0000    1.0000    1.0000     .9900    1.0000
PE ARRAY FOR I=   32
    .3200    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    .9900     .0700     .4700     .0100    0.0000    0.0000    0.0000
   1.0000    1.0000    1.0000     .0800     .6900     .2400    0.0000
    .9900    1.0000    1.0000    1.0000    1.0000     .9700     .8000
   1.0000    1.0000    1.0000    1.0000    1.0000     .9700    1.0000
   1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
   1.0000    1.0000     .9900    1.0000    1.0000     .9900     .9800
   1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
    .9900    1.0000    1.0000    1.0000    1.0000     .9800    1.0000
   1.0000    1.0000    1.0000    1.0000     .9900    1.0000     .9900
PE ARRAY FOR I=   33
    .4000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    .9700     .0100     .4700     .0100    0.0000    0.0000    0.0000
    .9700     .9600     .9500     .8800     .5700     .2800    0.0000
    .9600     .9500     .9700     .9800     .9600     .8800     .6200
    .9800     .9900     .9500     .9500     .9400     .9800     .9700
    .9700     .9500     .9600     .9900     .9600     .9700     .9600
    .9500     .9700     .9300     .9900     .9500     .9600     .9800
    .9600     .9900     .9500     .9500     .9500     .9400     .9600
    .9700     .9300     .9800     .9700     .9500     .9200     .9400
    .9800     .9600     .9800     .9600     .9700     .9600     .9700
PE ARRAY FOR I=   34
    .3400    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    .9600     .8600     .4000     .0100    0.0000    0.0000    0.0000
    .9600     .9500     .9400     .8800     .5700     .2500    0.0000
    .9400     .9600     .9800     .9200     .9500     .8400     .6800
    .9900     .9700     .9700     .9600     .9800     .9900     .9600
    .9800     .9500     .9300     .9800     .9500     .9500     .9600
    .9900     .9400     .9500     .9400     .9600     .9700     .9500
    .9400     .9800     .9700     .9500     .9800     .9800     .9700
    .9900     .9400     .9700     .9500     .9700     .9600     .9600
    .9700     .9600     .9700     .9700     .9500     .9500     .9800
PE ARRAY FOR I=   35
    .1900    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    .6900     .5600     .2600    0.0000    0.0000    0.0000    0.0000
    .6800     .6500     .7500     .5700     .2900     .1600    0.0000
    .7400     .7500     .7300     .7100     .6900     .5400     .2800
    .6800     .6600     .7600     .8000     .7200     .6700     .7100
    .6900     .7800     .6800     .7400     .7100     .7300     .7600
    .7000     .7300     .6500     .7100     .6600     .7100     .7800
    .7100     .6800     .7100     .7500     .7700     .7500     .7000
    .7000     .7400     .6600     .7300     .6700     .7300     .7500
    .7200     .6800     .6400     .7300     .7100     .7000     .7500
PE ARRAY FOR I=   36
    .1800    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    .6800     .5200     .2600    0.0000    0.0000    0.0000    0.0000
    .7100     .7800     .7800     .5300     .2600     .1100    0.0000
    .7000     .8000     .7200     .7400     .7200     .5700     .2200

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 10 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

     .6600     .7200     .7400     .7100     .7300     .7200     .6500
     .7100     .6800     .7300     .6900     .7000     .7800     .7100
     .7500     .7500     .6800     .7700     .6900     .6600     .7400
     .6700     .6500     .6900     .6700     .7000     .7600     .6500
     .7700     .8100     .6600     .7400     .6600     .7100     .7000
     .7300     .7100     .6900     .7100     .6900     .6500     .7300
PE ARRAY FOR I=   37
     .0600    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .3800     .2100     .0700     .0100    0.0000    0.0000    0.0000
     .3600     .3900     .3900     .2300     .0700     .0200    0.0000
     .4000     .3800     .3500     .4000     .3600     .3100     .1100
     .4200     .4300     .4100     .4500     .4200     .4100     .3000
     .4000     .4100     .3100     .4100     .3200     .3100     .4500
     .3400     .3800     .4200     .3700     .3400     .2900     .3200
     .4200     .2700     .3800     .2800     .3700     .4300     .4000
     .3600     .3100     .4600     .3800     .3600     .3800     .2800
     .3500     .3100     .4100     .3500     .4600     .3500     .2700
PE ARRAY FOR I=   38
     .1000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .3700     .2800     .0700    0.0000    0.0000    0.0000    0.0000
     .4700     .4200     .3300     .1600     .1100     .0200    0.0000
     .3300     .3800     .3400     .3500     .3700     .2300     .0800
     .3700     .4700     .3500     .3100     .3600     .3600     .3200
     .3300     .4500     .4200     .3100     .3300     .3800     .2600
     .3100     .4100     .3200     .3400     .4100     .3400     .3600
     .3600     .3800     .2600     .3400     .3500     .4600     .3900
     .3600     .3100     .3000     .3700     .3800     .3600     .3700
     .3700     .3500     .4100     .3700     .2800     .4000     .3900
PE ARRAY FOR I=   39
     .0100    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0500     .0400     .0100    0.0000    0.0000    0.0000    0.0000
     .0400     .0500     .0300     .0100    0.0000    0.0000    0.0000
     .0100     .1100     .0700     .0600     .0400     .0200    0.0000
     .0200     .0600     .0300     .0500     .0500     .0500     .0500
     .0100     .0800     .0200     .0400     .0600     .0500     .0400
     .0600     .0700     .0400     .0800     .0300     .0500     .0800
     .0500     .0200     .0300     .0400     .0300     .0500     .0600
     .0500     .0500     .0300     .0500     .0300     .0500     .0600
     .0500     .0100     .0700     .0200     .0400     .0800     .0500
PE ARRAY FOR I=   40
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0600     .0300     .0100    0.0000    0.0000    0.0000    0.0000
     .0800     .0300     .0700     .0400    0.0000     .0100    0.0000
     .1100     .0300     .0200     .0300     .0600     .0500     .0200
     .0300     .0600     .0700     .0300     .0400     .0400     .0200
     .0800     .0500     .0200     .0400     .0600     .0400     .0400
     .0500     .0600     .0800     .0200     .0400     .0400     .0400
     .0100     .0700     .0300     .0200     .0500     .0500     .0500
     .0400     .0100     .0700     .0800     .0200     .0100     .0400
     .0300     .0200     .0600     .0300     .0200     .0400     .0400

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 11 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

PE ARRAY FOR I=   41
     .2900   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .9100    .8900     .5000    0.0000    0.0000    0.0000    0.0000
     .8800    .8700     .9400     .8500     .6600     .1900    0.0000
     .8900    .8800     .9300     .8400     .8700     .8700     .7000
     .8900    .9000     .8900     .9400     .8800     .9300     .8700
     .8900    .8900     .8200     .9300     .9000     .9000     .9000
     .9100    .8800     .9000     .8400     .8300     .9300     .8400
     .8800    .9300     .8500     .9300     .9400     .8800     .9100
     .9000    .8600     .8700     .8400     .8700     .8500     .9100
     .9200    .9000     .8900     .8500     .8800     .8700     .9200
PE ARRAY FOR I=   42
     .3200   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .9300    .8900     .4700     .0100    0.0000    0.0000    0.0000
     .9300    .9100     .9100     .9500     .6900     .2400    0.0000
     .9200    .9100     .9500     .9300     .9100     .9000     .8000
     .9100    .9300     .9200     .9400     .9100     .9300     .9300
     .9400    .9100     .9400     .9300     .9200     .8400     .9000
     .9100    .9600     .9100     .9300     .9500     .9600     .9100
     .9100    .9400     .9000     .8800     .9100     .9500     .9400
     .9100    .9100     .9200     .8800     .9500     .8900     .9600
     .9100    .9100     .9300     .9500     .9100     .9700     .9400
PE ARRAY FOR I=   43
     .4000   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .6000    .7700     .4700     .0100    0.0000    0.0000    0.0000
     .7100    .7400     .6700     .7700     .5700     .2800    0.0000
     .6900    .6800     .6700     .6900     .6900     .7700     .6200
     .7100    .6500     .6800     .7400     .6500     .7000     .7800
     .7200    .6500     .7100     .6200     .7000     .8200     .7200
     .7100    .7300     .7700     .7500     .7100     .7300     .6800
     .5800    .8000     .6900     .6300     .7000     .6300     .6600
     .7000    .5700     .6700     .6500     .7300     .6700     .6700
     .7400    .7600     .6700     .6700     .6700     .7700     .6800
PE ARRAY FOR I=   44
     .3400   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .6800    .6600     .4000     .0100    0.0000    0.0000    0.0000
     .7500    .7100     .6400     .7300     .5700     .2500    0.0000
     .6800    .7100     .8000     .7000     .7000     .7200     .6800
     .7400    .6700     .7600     .6900     .7900     .7600     .7000
     .6500    .6600     .7000     .7100     .7200     .7400     .7100
     .7500    .6200     .7700     .7100     .7400     .6300     .7100
     .6900    .8100     .6600     .6200     .7600     .7300     .7200
     .7200    .7000     .6600     .7600     .7400     .7400     .6700
     .6900    .6500     .7300     .7400     .6400     .7600     .7000
PE ARRAY FOR I=   45
     .1900   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .2900    .2000     .2600    0.0000    0.0000    0.0000    0.0000
     .2700    .2600     .3200     .3300     .2900     .1600    0.0000
     .3600    .2900     .3200     .2600     .2200     .2900     .2800
     .3700    .3200     .3900     .3600     .3400     .2800     .2700

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 12 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

      .2800     .2900     .3400     .3900     .2700     .3300     .3400
      .2800     .2800     .3100     .2700     .2300     .2500     .3400
      .2800     .3200     .2600     .3200     .3800     .2900     .2400
      .2700     .2500     .2400     .2900     .2800     .3100     .2200
      .2500     .3400     .2900     .3600     .2900     .3300     .3000
PF ARRAY FOR I=  46
      .1800    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .2500     .2700     .2600    0.0000    0.0000    0.0000    0.0000
      .2600     .3600     .3200     .2900     .2600     .1100    0.0000
      .3100     .3800     .3600     .2200     .3200     .2700     .2200
      .3300     .3000     .2900     .3200     .3000     .2900     .1900
      .2900     .2800     .2700     .2700     .2900     .2800     .2800
      .3000     .2600     .2700     .3800     .3300     .2500     .2300
      .2700     .2500     .2800     .3400     .2500     .2900     .2800
      .3600     .4000     .2200     .3000     .2100     .3600     .2300
      .3100     .2900     .2600     .2900     .3100     .2700     .2600
PF ARRAY FOR I=  47
      .0600    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .1400     .0600     .0700     .0100    0.0000    0.0000    0.0000
      .1500     .1000     .0800     .1000     .0700     .0200    0.0000
      .1000     .1100     .1000     .1200     .1300     .1100     .1100
      .0900     .0700     .1800     .1500     .0800     .1500     .0700
      .1200     .1100     .0500     .1300     .1400     .0700     .0800
      .1400     .0900     .1000     .1100     .1100     .0800     .0700
      .0400     .0900     .0500     .0800     .0800     .1200     .1100
      .0700     .1200     .1400     .1200     .0600     .1000     .0700
      .1200     .0600     .1600     .1000     .1500     .1100     .0600
PF ARRAY FOR I=  48
      .1000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
      .0900     .1800     .0700    0.0000    0.0000    0.0000    0.0000
      .1400     .1400     .0900     .0900     .1100     .0200    0.0000
      .1000     .0700     .1300     .1000     .0900     .0800     .0800
      .1000     .1000     .0800     .1200     .1000     .1100     .0700
      .0500     .1000     .1300     .1200     .0400     .1200     .0500
      .0400     .0800     .0300     .0700     .0900     .1000     .0700
      .1000     .1300     .0700     .0500     .0900     .1600     .0900
      .1200     .0300     .0700     .0600     .0700     .1300     .1200
      .1200     .0700     .1000     .0600     .0700     .0900     .0600
PF ARRAY FOR I=  49
      .0100    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     0.0000    0.0000     .0100    0.0000    0.0000    0.0000    0.0000
      .0200     .0200     .0200    0.0000    0.0000    0.0000    0.0000
      .0100     .0100     .0200    0.0000    0.0000    0.0000    0.0000
     0.0000     .0200    0.0000     .0100    0.0000     .0100     .0100
     0.0000     .0100    0.0000     .0100     .0100     .0200    0.0000
      .0100     .0100     .0100     .0100     .0100     .0100     .0100
      .0100    0.0000     .0100    0.0000     .0100     .0200     .0100
      .0200    0.0000    0.0000    0.0000    0.0000     .0100     .0100
     0.0000    0.0000     .0100     .0100    0.0000     .0100    0.0000
PF ARRAY FOR I=  50

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 13 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000     .0100     .0100    0.0000    0.0000    0.0000    0.0000
     .0100     .0200     .0100     .0100    0.0000     .0100    0.0000
     .0200     .0100    0.0000    0.0000     .0200     .0100     .0200
    0.0000    0.0000     .0200     .0100     .0200     .0100     .0100
     .0100     .0200    0.0000    0.0000     .0200    0.0000     .0100
     .0100     .0200    0.0000    0.0000     .0100     .0100    0.0000
    0.0000    0.0000    0.0000    0.0000     .0100    0.0000     .0100
     .0100    0.0000     .0100     .0300    0.0000    0.0000    0.0000
    0.0000     .0200     .0300    0.0000     .0100     .0300     .0100
PF ARRAY FOR I=   51
     .2900    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .9100     .8900     .5000    0.0000    0.0000    0.0000    0.0000
     .8800     .8700     .9400     .8500     .6600     .1900    0.0000
     .8900     .8800     .9300     .8400     .8700     .8700     .7000
     .8900     .9000     .8900     .9400     .8800     .9300     .8700
     .8900     .8900     .8200     .9300     .9000     .9000     .9000
     .9100     .8800     .9000     .8400     .8300     .9300     .8400
     .8800     .9300     .8500     .9300     .9400     .8800     .9100
     .9000     .8600     .8700     .8400     .8700     .8500     .9100
     .9200     .9000     .8900     .8500     .8800     .8700     .9200
PF ARRAY FOR I=   52
     .3200    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .9300     .8900     .4700     .0100    0.0000    0.0000    0.0000
     .9300     .9100     .9100     .9500     .6900     .2400    0.0000
     .9200     .9100     .9500     .9300     .9100     .9000     .8000
     .9100     .9300     .9200     .9400     .9100     .9300     .9300
     .9400     .9100     .9400     .9300     .9200     .8400     .9000
     .9100     .9600     .9100     .9300     .9500     .9600     .9100
     .9100     .9400     .9000     .8800     .9100     .9500     .9400
     .9100     .9100     .9200     .8800     .9500     .8900     .9600
     .9100     .9100     .9300     .9500     .9100     .9700     .9400
PF ARRAY FOR I=   53
     .4000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .6000     .7700     .4700     .0100    0.0000    0.0000    0.0000
     .7100     .7400     .6700     .7700     .5700     .2800    0.0000
     .6900     .6800     .6700     .6900     .6900     .7700     .6200
     .7100     .6500     .6800     .7400     .6500     .7000     .7800
     .7200     .6500     .7100     .6200     .7000     .8200     .7200
     .7100     .7300     .7700     .7500     .7100     .7300     .6800
     .5800     .8000     .6900     .6300     .7000     .6300     .6600
     .7000     .5700     .6700     .6500     .7300     .6700     .6700
     .7400     .7600     .6700     .6700     .6700     .7700     .6800
PF ARRAY FOR I=   54
     .3400    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .6800     .6600     .4000     .0100    0.0000    0.0000    0.0000
     .7500     .7100     .6400     .7300     .5700     .2500    0.0000
     .6800     .7100     .8000     .7000     .7000     .7200     .6800
     .7400     .6700     .7600     .6900     .7200     .7600     .7900
     .6500     .6600     .7000     .7100     .7200     .7400     .7100

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2  LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 14 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

     .7500     .6200     .7700     .7100     .7400     .6300     .7100
     .6900     .8100     .6600     .6200     .7600     .7300     .7200
     .7200     .7000     .6600     .7600     .7400     .7400     .6700
     .6900     .6500     .7300     .7400     .6400     .7600     .7000
PF ARRAY FOR I=   55
     .1900    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .2900     .2000     .2600    0.0000    0.0000    0.0000    0.0000
     .2700     .2600     .3200     .3300     .2900     .1600    0.0000
     .3600     .2900     .3200     .2600     .2200     .2900     .2800
     .3200     .3200     .3700     .3600     .3400     .2800     .2700
     .2800     .2900     .3400     .3900     .2700     .3300     .3400
     .2800     .2800     .3100     .2700     .2300     .2500     .3400
     .2800     .3200     .2600     .3200     .3800     .2900     .2400
     .2700     .2500     .2400     .2900     .2800     .3100     .2200
     .2500     .3400     .2900     .3600     .2900     .3300     .3000
PF ARRAY FOR I=   56
     .1800    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .2500     .2700     .2600    0.0000    0.0000    0.0000    0.0000
     .2600     .3600     .3200     .2900     .2600     .1100    0.0000
     .3100     .3800     .3600     .2200     .3200     .2700     .2200
     .3300     .3000     .2900     .3200     .3000     .2900     .1900
     .2900     .2800     .2700     .2700     .2900     .2800     .2800
     .3000     .2600     .2700     .3800     .3300     .2500     .2300
     .2700     .2500     .2800     .3400     .2500     .2900     .2800
     .3600     .4000     .2200     .3000     .2100     .3600     .2300
     .3100     .2900     .2600     .2900     .3100     .2700     .2600
PF ARRAY FOR I=   57
     .0600    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .1400     .0600     .0700     .0100    0.0000    0.0000    0.0000
     .1500     .1000     .0800     .1000     .0700     .0200    0.0000
     .1000     .1100     .1000     .1200     .1300     .1100     .1100
     .0900     .0700     .1800     .1500     .0800     .1500     .0700
     .1200     .1100     .0500     .1300     .1400     .0700     .0800
     .1400     .0900     .1000     .1100     .1100     .0800     .0700
     .0400     .0900     .0500     .0800     .0800     .1200     .1100
     .0700     .1200     .1400     .1200     .0600     .1000     .0700
     .1200     .0600     .1600     .1000     .1500     .1100     .0600
PF ARRAY FOR I=   58
     .1000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0900     .1800     .0700    0.0000    0.0000    0.0000    0.0000
     .1400     .1400     .0900     .0900     .1100     .0200    0.0000
     .1000     .0700     .1300     .1000     .0900     .0800     .0800
     .1000     .1000     .0800     .1200     .1000     .1100     .0700
     .0500     .1000     .1300     .1200     .0400     .1200     .0500
     .0400     .0800     .0300     .0900     .0900     .1000     .0700
     .1000     .1300     .0700     .0500     .0900     .1600     .0900
     .1200     .0300     .0700     .0600     .0700     .1300     .1200
     .1200     .0700     .1000     .0600     .0700     .0900     .0600
PF ARRAY FOR I=   59
     .0100    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 15 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

     0.0000     0.0000      .0100     0.0000     0.0000     0.0000     0.0000
      .0200      .0200      .0200     0.0000     0.0000     0.0000     0.0000
      .0100      .0100      .0200     0.0000     0.0000     0.0000     0.0000
     0.0000      .0200     0.0000      .0100     0.0000      .0100      .0100
     0.0000      .0100     0.0000      .0100      .0100      .0200     0.0000
      .0100      .0100      .0100      .0100      .0100      .0100      .0100
      .0100     0.0000      .0100     0.0000      .0100      .0200      .0100
      .0200     0.0000     0.0000     0.0000     0.0000      .0100      .0100
     0.0000     0.0000      .0100      .0100     0.0000      .0100     0.0000
PE ARRAY FOR I=   60
     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000
     0.0000      .0100      .0100     0.0000     0.0000     0.0000     0.0000
      .0100      .0200      .0100      .0100     0.0000      .0100     0.0000
      .0200      .0100     0.0000     0.0000      .0200      .0100      .0200
     0.0000     0.0000      .0200      .0100      .0200      .0100      .0100
      .0100      .0200     0.0000     0.0000      .0200     0.0000      .0100
      .0100      .0200     0.0000     0.0000      .0100      .0100     0.0000
     0.0000     0.0000     0.0000     0.0000      .0100     0.0000      .0100
      .0100     0.0000      .0100      .0300     0.0000     0.0000     0.0000
     0.0000      .0200      .0300     0.0000      .0100      .0300      .0100
    16
     0.0000      .9900      .9500      .9900      .8000      .9000      .6000
   500.0000      .9500      .9000      .9500      .7500      .8500      .5500
  1000.0000      .9500      .8500      .9000      .7000      .8000      .4000
  1500.0000      .9000      .8000      .8500      .6000      .6000      .3000
  2000.0000      .8500      .8000      .8500      .6500      .7500      .3000
  2500.0000      .8000      .7500      .7000      .6000      .7000      .1500
  3000.0000      .7500      .7000      .6500      .5000      .6000      .0500
  3500.0000      .7000      .6500      .6000      .4500     0.0000     0.0000
  4000.0000      .6500      .6000      .5500      .4000     0.0000     0.0000
  4500.0000      .6000      .5500      .5000      .3500     0.0000     0.0000
  5000.0000      .5500      .5000      .4500      .3000     0.0000     0.0000
  5500.0000      .5000      .4500      .4000      .2500     0.0000     0.0000
  6000.0000      .4500      .4000      .3500      .2000     0.0000     0.0000
  6500.0000      .4000      .3500      .3000      .1500     0.0000     0.0000
  7000.0000      .3500      .3000      .2500      .1000     0.0000     0.0000
  9000.0000      .3000      .2000      .1000     0.0000     0.0000     0.0000
     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000
     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000
     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000
     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000     0.0000
     0.0000    30.0000    90.0000   150.0000   300.0000  9000.0000
      .5000      .7000      .5000      .8000      .7000      .9500      .6000     0.0000     0.0000     0.0000
      .5000      .6000      .4500      .7500      .7000     1.0000      .3000     0.0000     0.0000     0.0000
     8
     0.0000   500.0000  1000.0000  1500.0000  2000.0000  2500.0000  3000.0000  9000.0000     0.0000     0.0000
      .0700      .0700      .1300      .1500      .1600      .1700      .1800      .1800     0.0000     0.0000
      .9000      .9000      .8200      .7800      .7600      .7400      .7300      .7300     0.0000     0.0000
      .0300      .0300      .0500      .0700      .0100      .0900      .0900      .0900     0.0000     0.0000
    20.0000    20.0000    20.0000     0.0000     0.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING 7-2 LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 16 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  69.1900   69.1900   69.1900   69.1900   46.3200   46.3200   46.3200   46.3200
   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  9
   0.0000     .1250     .2500     .3750     .5000     .6250     .7500     .8750    1.0000    0.0000
   0.0000    5.0000   10.0000   18.0000   24.0000   36.0000   52.0000   87.0000  174.0000    0.0000
  75.0000   75.0000   75.0000   75.0000   75.0000   75.0000
 150.0000  181.0000     .9150     .9750     .9750     .9750    0.0000
  2
********************************************* END OF DATA CHECK FOR CASE   2 ****************************************
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

***********         SANDM4R    0011110 LINES PRINTED.  (LS10)
***********         SANDM4R    0011110 LINES PRINTED.  (LS10)
```

LISTING 7-2  LONG FORM OF ECHO PRINTS - CONT'D

(PAGE 17 OF 17)

```
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM   DEURS   MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM   08/18/80
                                                                                                                      08
```

CHAPTER 8

# 8. PROGRAM DESIGN

## 8.1 Programming Principles of COPE.

The main COPE program has been designed according to the following principles:

(1) Input is designed to allow multiple cases per run with only a few mnemonically keyed input lines per case. For flexibility, it is also possible to read in data as traditional FORTRAN formatted input.

(2) Output is designed so that the printout of results is nearly self-explanatory. In addition, user generated comments can easily be made to appear in the printout for any case.

(3) The program is highly modularized. The main program consists almost entirely of calls to various subroutines together with the logic necessary to initialize and control the execution of three nested loops (see section 8.2).

(4) Most subroutines or functions are designed for a single purpose (e.g., perform a test, read input, write output, etc.) so that when one wishes to change any aspect of the program it is usually necessary to change only a small number of the subroutines (usually the one performing a particular test plus the input routine).

(5) The principles of structured programming have been adhered to in some degree. Because the program is written in FORTRAN IV (extended CDC version) which does not have if-then-else, do-until, do-while constructs, it is still necessary to use GO TO's and statement labels. However, blank comment cards have been inserted between logical blocks within each program to improve readability, and unnecessary GO TO's have been minimized. In addition, statement labels are in ascending numerical order within each subroutine.

(6) All data statements have been grouped into a single BLOCK DATA subprogram.

(7) Mnemonically meaningful variable and subroutine names have been selected.

(8) FORTRAN code has been written to minimize the number of non-standard (i.e., CDC peculiar) features used. Nevertheless, there are some CDC peculiar code features that were impractical to omit (e.g., READMS, SYSTEMC, etc.). These are cited in Chapter 13.

## 8.2 General Logic Flow.

The COPE main program can be viewed as three large nested loops on case, replication, and round. Figure 8-1 illustrates the general COPE logic flow including the three loops.

This figure is necessarily oversimplified and a more detailed flow chart is to be found in Chapter 11, but the overall picture is accurately summed up here.

## 8.3 Subroutine and Function Calling.

Table 8-2 gives a listing of the names of the main program, all of the subroutines, and all of the functions in the COPE program.

COPE is the main program name.

FPRCNT, GAMMA, IDCHAR, NUMRIC, and URAN31 are the function names.

ABRTTL, ADDTBR, ADDTOF, ARTCHK, BLOCHK, CCPLOT, COMMNT, CREAD, DETCTN, DFAULT, DFCHK, DUST, ECHO, FINDIT, GETRNG, GETTIM, GETVIS, HEADER, HITCHK, INITLZ, INPUT, LOSCHK, MINTRN, NOREC, OUTPUT, PECHK, PENAME, PKCHK, PPLOT, PSMOKE, REINTZ, RNDREL, RNGCHK, SEPREC, SMOKE, SMPLCD, TIMCHK, TITLE, USET, VISCHK, and WARNFO are the subroutine names.

PEIDNT is an entry point name in subroutine PENAME.

For each list entry (subprogram name) in Table 8-2, there are sections giving the names of subprograms calling the given list entry, the names of subprograms (in COPE) that the given list entry calls, and the names of system or IMSL routines that the given list entry calls.

There is a discrepancy that occurs in the list. Subroutine NOREC is called by system routines LOCF (as an external argument) and SYSTEMC (by address loaded in IRAY array) and hence does not show as "CALLED BY" any routine in Figure 8-2 (see section 9.31).

Table 8-3 lists the system (CDC, standard FORTRAN, and International Mathematical and Statistical Library) routines used in COPE and the subprograms of COPE that call each of them.

Table 8-4 is a subprogram calling tree. Each subprogram (subroutine or function) in column two can be called by the COPE main program directly. Each subprogram in column N calls those subprograms in column N+1 to which it is linked by lines (V's, I's, -'s, and >'s).

NOREC calls no other subprograms in COPE. It is called by SYSTEMC whenever CDC FORTRAN IV Extended execution error 104 is encountered in INPUT, INITLZ, or FINDIT.

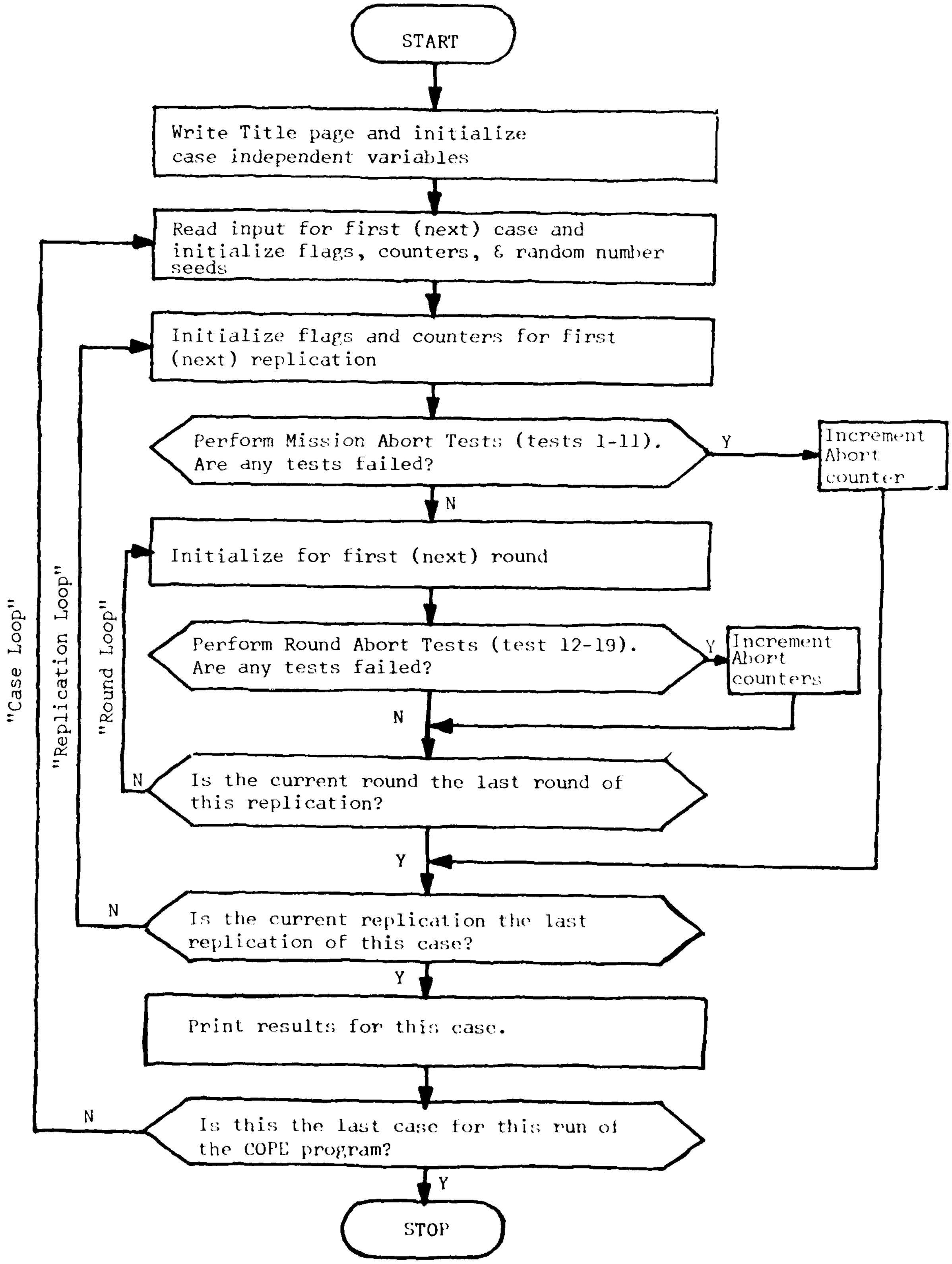



FIGURE 8-1 General COPE Logic Flow

TABLE 8-2 PROGRAM UNIT REFERENCES (SUBROUTINES, FUNCTIONS, AND MAIN PROGRAM)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ABRTTL | CALLED BY | ARTCHK | BLOCHK | DETCTN | DFCHK | DUST | GETTIM |
| | | HITCHK | LOSCHK | MINTRN | PECHK | PKCHK | RNDREL |
| | | RNGCHK | SMOKE | TIMCHK | VISCHK | WARNDO | |
| ADDTBR | CALLED BY | COPE | | | | | |
| ADDTOF | CALLED BY | COPE | | | | | |
| ARTCHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| BLOCHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | | | | | |
| CCPLOT | CALLED BY | INPUT | | | | | |
| | CALLS | COMMNT | | | | | |
| | CALLS (SYS) | ENCODE | | | | | |
| COMMNT | CALLED BY | CCPLOT | HEADER | INPUT | PPLOT | SEPREC | |
| | CALLS (SYS) | ENCODE | WRITE | | | | |
| COPE | CALLS | ADDTBR | ADDTOF | ARTCHK | BLOCHK | DETCTN | DFCHK |
| | | DUST | GETRNG | GETTIM | GETVIS | HITCHK | INITLZ |
| | | INPUT | LOSCHK | MINTRN | OUTPUT | PECHK | PKCHK |
| | | REINTZ | RNDREL | RNGCHK | SMOKE | TIMCHK | TITLE |
| | | VISCHK | WARNDO | | | | |
| CREAD | CALLED BY | INPUT | | | | | |
| | CALLS | SEPREC | | | | | |
| | CALLS (SYS) | DECODE | ENCODE | READ | | | |
| DETCTN | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | SMPLCD | | | | |
| DFAULT | CALLED BY | INITLZ | | | | | |
| DFCHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| | CALLS (SYS) | AMAX1 | | | | | |
| DUST | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| ECHO | CALLED BY | INPUT | | | | | |
| | CALLS (SYS) | WRITE | | | | | |
| FINDIT | CALLED BY | INPUT | | | | | |
| | CALLS | NUMRIC | | | | | |
| | CALLS (SYS) | IFIX | LOCF | READMS | SYSTEMC | WRITE | |
| FPRCNT | CALLED BY | OUTPUT | | | | | |
| | CALLS (SYS) | ENCODE | | | | | |
| GAMMA | CALLED BY | GETTIM | | | | | |
| | CALLS (IMSL) | GGAMA | | | | | |

TABLE 8-2 PROGRAM UNIT REFERENCES (SUBROUTINES, FUNCTIONS, AND MAIN PROGRAM) - CONT'D

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| GETRNG | CALLED BY | COPE | | | | | |
| | CALLS | SMPLCD | | | | | |
| | CALLS (SYS) | FLOAT | STRACE | | | | |
| GETTIM | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | GAMMA | SMPLCD | URAN31 | | |
| GETVIS | CALLED BY | COPE | | | | | |
| | CALLS | URAN31 | | | | | |
| | CALLS (SYS) | STRACE | | | | | |
| HEADER | CALLED BY | INPUT | | | | | |
| | CALLS | COMMNT | PEIDNT | | | | |
| | CALLS (SYS) | DECODE | ENCODE | WRITE | | | |
| HITCHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| IDCHAR | CALLED BY | NUMRIC | SEPREC | | | | |
| INITLZ | CALLED BY | COPE | | | | | |
| | CALLS | DFAULT | USET | | | | |
| | CALLS (SYS) | LOCF | OPENMS | READMS | SYSTEMC | | |
| INPUT | CALLED BY | COPE | | | | | |
| | CALLS | CCPLOT | COMMNT | CREAD | ECHO | FINDIT | HEADER |
| | | PENAME | PPLOT | PSMOKE | SEPREC | USET | |
| | CALLS (SYS) | ABS | DECODE | ENCODE | EOF | FLOAT | LOCF |
| | | READMS | READ | SYSTEMC | WRITE | | |
| LOSCHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| | CALLS (SYS) | AMAX1 | | | | | |
| MINTRN | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| | CALLS (SYS) | AMAX1 | | | | | |
| NOREC | CALLED BY | | | | | | |
| | CALLS (SYS) | STRACE | WRITE | | | | |
| NUMRIC | CALLED BY | FINDIT | | | | | |
| | CALLS | IDCHAR | | | | | |
| | CALLS (SYS) | DECODE | ENCODE | WRITE | | | |
| OUTPUT | CALLED BY | COPE | | | | | |
| | CALLS | PPRCNT | | | | | |
| | CALLS (SYS) | FLOAT | WRITE | | | | |
| PECHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| | CALLS (SYS) | AMAX1 | FLOAT | MAXØ | MINØ | MIN1 | |
| PEIDNT | CALLED BY | HEADER | | | | | |
| | CALLS (SYS) | ABS | DECODE | FLOAT | MOD | | |
| PENAME | CALLED BY | INPUT | | | | | |
| | CALLS (SYS) | ABS | ENCODE | | | | |

TABLE 8-2 PROGRAM UNIT REFERENCES (SUBROUTINES, FUNCTIONS, AND MAIN PROGRAM) - CONT'D

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| PKCHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| | CALLS (SYS) | MIN0 | | | | | |
| PPLOT | CALLED BY | INPUT | | | | | |
| | CALLS | COMMNT | | | | | |
| | CALLS (SYS) | ENCODE | | | | | |
| PSMOKE | CALLED BY | INPUT | | | | | |
| | CALLS (SYS) | AMIN1 | FLOAT | | | | |
| REINTZ | CALLED BY | COPE | | | | | |
| | CALLS (SYS) | MOD | WRITE | | | | |
| RNDREL | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| RNGCHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | | | | | |
| SEPREC | CALLED BY | CREAD | INPUT | | | | |
| | CALLS | COMMNT | IDCHAR | | | | |
| | CALLS (SYS) | ENCODE | | | | | |
| SMOKE | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |
| SMPLCD | CALLED BY | DETCTN | GETRNG | GETTIM | | | |
| | CALLS | URAN31 | | | | | |
| | CALLS (SYS) | STRACE | WRITE | | | | |
| TIMCHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | | | | | |
| TITLE | CALLED BY | COPE | | | | | |
| | CALLS (SYS) | DATE | TIME | WRITE | | | |
| URAN31 | CALLED BY | ARTCHK | DFCHK | DUST | GETTIM | GETVIS | HITCHK |
| | | LOSCHK | MINTRN | PECHK | PKCHK | RNDREL | SMOKE |
| | | SMPLCD | WARNDO | | | | |
| USET | CALLED BY | INITLZ | INPUT | | | | |
| VISCHK | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | | | | | |
| WARNDO | CALLED BY | COPE | | | | | |
| | CALLS | ABRTTL | URAN31 | | | | |

TABLE 8-3 SYSTEM AND IMSL ROUTINE REFERENCES

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ABS | CALLED BY | INPUT | PEIDNT | PENAME | | | |
| AMAX1 | CALLED BY | DECHK | LOSCHK | MINTRN | PECHK | | |
| AMIN1 | CALLED BY | PSMOKE | | | | | |
| DATE | CALLED BY | TITLE | | | | | |
| DECODE | CALLED BY | CREAD | HEADER | INPUT | NUMRIC | PEIDNT | |
| ENCODE | CALLED BY | CCPLOT | COMMNT | CREAD | ERRCND | HEADER | INPUT |
| | | NUMRIC | PENAME | PPLOT | SEPREC | | |
| EOF | CALLED BY | INPUT | | | | | |
| FLOAT | CALLED BY | GETRNG | INPUT | OUTPUT | PECHK | PEIDNT | PSMOKE |
| GGAMA | CALLED BY | GAMMA | | | | | |
| IFIX | CALLED BY | FINDIT | | | | | |
| LOCF | CALLED BY | FINDIT | INITLZ | INPUT | | | |
| MAX0 | CALLED BY | PECHK | | | | | |
| MIN0 | CALLED BY | PECHK | PRCHK | | | | |
| MIN1 | CALLED BY | PECHK | | | | | |
| MOD | CALLED BY | PEIDNT | REINTZ | | | | |
| OPENMS | CALLED BY | INITLZ | | | | | |
| READMS | CALLED BY | FINDIT | INITLZ | INPUT | | | |
| READ | CALLED BY | CREAD | INPUT | | | | |
| STRACE | CALLED BY | GETRNG | GETVIS | NOREC | SMPLCD | | |
| SYSTEMC | CALLED BY | FINDIT | INITLZ | INPUT | | | |
| TIME | CALLED BY | TITLE | | | | | |
| WRITE | CALLED BY | COMMNT | ECHO | FINDIT | HEADER | INPUT | NOREC |
| | | NUMRIC | OUTPUT | REINTZ | SMPLCD | TITLE | |

NOTE: ALL ROUTINES LISTED IN THE FIRST COLUMN OF THIS TABLE (EXCEPT GGAMA) ARE STANDARD SYSTEM ROUTINES OF CDC FORTRAN EXTENDED VERSION 4.
GGAMA IS A SUBROUTINE FROM THE INTERNATIONAL MATHEMATICAL AND STATISTICAL LIBRARY (IMSL).

TABLE 8-4 COPE SUBROUTINE "TREE"

```
COPE------->ADDTBR
    I------>ADDTOF
    I------>ARTCHK------>ABRTTL
    V             I----->URAN31
    I------>BLOCHK------>ABRTTL
    I------>DETCTN------>ABRTTL
    V             I----->SMPLCD------>URAN31
    I------>DFCHK------->ABRTTL
    V             I----->URAN31
    I------>DUST-------->ABRTTL
    V             I----->URAN31
    I------>GETRNG------>SMPLCD------>URAN31
    I------>GETTIM------>ABRTTL
    V             I----->GAMMA
    V             I----->SMPLCD------>URAN31
    V             I----->URAN31
    I------>GETVIS------>URAN31
    I------>HITCHK------>ABRTTL
    V             I----->URAN31
    I------>INITLZ------>DFAULT
    V             I----->USET
    I------>INPUT------->CCPLOT------>COMMNT
    V             I----->COMMNT
    V             I----->CREAD------->SEPREC------>COMMNT
    V             V                         I----->IDCHAR
    V             I----->ECHO
    V             I----->FINDIT------>NUMRIC------>IDCHAR
    V             I----->HEADER------>COMMNT
    V             V            I----->PEIDNT
    V             I----->PENAME
    V             I----->PPLOT------->COMMNT
    V             I----->PSMOKE
    V             I----->SEPREC------>COMMNT
    V             V            I----->IDCHAR
    V             I----->USET
    I------>LOSCHK------>ABRTTL
    V             I----->URAN31
    I------>MINTRN------>ABRTTL
    V             I----->URAN31
    I------>OUTPUT------>FPRCNT
    I------>PECHK------->ABRTTL
    V             I----->URAN31
    I------>PKCHK------->ABRTTL
    V             I----->URAN31
    I------>REINTZ
    I------>RNDREL------>ABRTTL
    V             I----->URAN31
    I------>RNGCHK------>ABRTTL
    I------>SMOKE------->ABRTTL
    V             I----->URAN31
    I------>TIMCHK------>ABRTTL
    I------>TITLE
    I------>VISCHK------>ABRTTL
    I------>WARNDO------>ABRTTL
    V             I----->URAN31
NOREC
```

CHAPTER 9

# 9. SUBPROGRAMS

This chapter explains in moderate detail what each subprogram of COPE does. All four standard FORTRAN subprogram types (main program, block data, function, and subroutine) are present in COPE. In the following, the subprograms are discussed in this order: main program, block data, functions (in alphabetical order), and subroutines (in alphabetical order).

This chapter deals only with subprograms defined by COPE. System subroutines and functions are divided into three classes: standard FORTRAN IV routines, routines peculiar to CDC FORTRAN 4 Extended, and IMSL routines. The reader is assumed to be familiar with the first of these classes (or at least have access to a FORTRAN manual). System routines in the remaining two classes are discussed to some extent in Chapter 13.

## 9.1 COPE Main Program.

The COPE main program is well summarized by the flowchart in Figure 8-1.

The main program begins by calling routines that write the title page (TITLE) and initialize case independent variables (INITLZ).

Next the main program calls INPUT to read the first (or next) case's input. It then calls REINTZ which re-initializes certain case dependent counters and random number seeds. It then initializes the current replication number at one and the number of FO's (DO's) killed so far in the case at zero. The steps in this paragraph begin the "case loop."

The main program next begins the "replication loop" for the first (or next) replication of the current case. This loop begins by initializing the number of the current round to one and then setting various times, flags, and a counter.

The "replication loop" now continues by calling a sequence of subroutines which perform the tests described in Chapter 2 and sample weather, range-LOS, and time data.

| routines (in order called) | purpose |
|---|---|
| ARTCHK | perform test 1 |
| GETVIS | sample weather distribution |
| GETRNG | sample acquisition range and LOS segment distributions |
| VISCHK | perform test 2 |
| RNGCHK | perform test 3 |
| DETCTN | perform test 4 |
| SMOKE | perform test 5 |
| DUST | perform test 6 |
| BLOCHK | perform test 7 |
| GETTIM | sample time distribution(s) and perform tests 8 and 9 |
| BLOCHK | perform test 10 |
| TIMCHK | perform test 11 (for "shooting gallery" only) |
| ADDTOF | calculate round arrival time |

Each of these routines is explained in detail later in this chapter.

Note that if any routine that performs one (or more) of the tests is called and the test is failed, then the subsequent routines in the above list are not called; instead, control transfers to statement 140 (end of "replication loop") where a check is made to determine whether all of the replications for this case have been done. If they have, OUTPUT is called to print case results and then the end of the "case loop" is either hit and control returns to statement 100 which calls INPUT to read the input for the next case or, if the last case has been run, control passes to the STOP and the program terminates. If not all replications for the current case have been completed, then the number of the current replication is increased by one and control returns to statement 110 (which is the start of the "replication loop") to begin next replication.

If none of the tests are failed and the program reaches ADDTOF on the current replication, then the "round loop" begins (with statement 120). The "round loop" consists of a further sequence of subroutines which are called to perform those tests that may affect only one round.

| routines(in order called) | purpose |
|---|---|
| LOSCHK | perform test 12 |
| WARNFO | perform test 13 |
| DFCHK | perform test 14 |
| RNDREL | perform test 15 |
| MINTRN | perform test 16 |
| PECHK | perform test 17 |
| HITCHK | perform test 18 |
| PKCHK | perform test 19 |

Again, each of these routines is explained in detail later in this chapter.

If any subroutine that performs one of these tests (12 through 19) is called and the test is failed, then the subsequent routines in the above list are not called for the current round; instead, control transfers to statement 130 (end of "round loop") where a check is made to determine whether all of the rounds for this replication have been done. If they have, control transfers to statement 140 (end of "replication loop") where the check is made to determine whether more replications are to be done for the current case. If not all rounds for this replication have been completed, then the time the next round arrives is computed (ADDTBR), the round number is incremented by one, the end replication (or ENDREP) flag is reset and control transfers to statement 120 (start of "round loop") to begin the next round.

If no test is failed, then control reaches statement 130 after the entire sequence of subroutines in the "round loop" has been called. At this point a check is made to determine whether all rounds for this replication have been done. Control then proceeds just as in the case when statement 130 is reached due to a failed test as described in the paragraph above.

When all the rounds for a replication are completed, a check is made to determine whether more replications remain to be done (end of "replication loop"). If more replications remain, control transfers to 110 (start of "replication loop") to begin the next replication. If all replications for the current case are completed, then OUTPUT is called to print results for current case (end of "case loop").

When a case ends and its results are printed, the program checks whether the last case flag (ENDFLG) has been set. If it has, the program stops; if it has not been set, then control goes to statement 100 and INPUT is called to read the next cases input. If there is input for another case, it is read and the entire process from INPUT through end of "case loop" is repeated; if there is no input for further cases, the program terminates.

## 9.2 BDATA1 Block Data Subprogram.

BDATA1 consists entirely of data statements used to initialize variables in various COMMON blocks. These blocks are listed in section 4.7 and are their variables defined individually in Chapter 12 (Glossary of Variables).

## 9.3 FPRCNT Function.

This function formats percentages. It takes a numerical argument XIN and a logical argument PAREN and assigns a one word alphanumeric character string as the value of FPRCNT.

If XIN is negative, then FPRCNT is assigned:
the left justified character string UNDEF), when PAREN is true;
the right justified character string UNDEF, when PAREN is false.

If XIN $\geq$ 100.05, a STOP occurs with an error message (see section 7.5.7).

If XIN is positive and less than 100.05, then FPRCNT is assigned:

a left justified character string of the form X.XX%), XX.XX%), or 100.0%), when PAREN is true;

a right justified character string of the form X.XX%, XX.XX%, or 100.0%, when PAREN is false.

The character strings created by FPRCNT are used by OUTPUT to print percentages. FPRCNT is called only from OUTPUT.

## 9.4 GAMMA Function.

This function takes two arguments, one real parameter ALPHA and one integer IRN, and returns a real value as GAMMA.

ALPHA is the $\alpha$ -parameter of the gamma distribution's probability density function:

$$g(x) = \frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x}$$

GAMMA is assigned a value obtained by randomly sampling from the above distribution. This sampling is accomplished by calling the IMSL subroutine GGAMA with random number seed IRN.

The function GAMMA itself then really does nothing more than dimension two arrays required by GGAMA, define an additional variable used by GGAMA, call GGAMA with ALPHA, IRN, the two arrays and the additional variable as arguments, and then set GAMMA equal to the random sample obtained by GGAMA from the gamma distribution.

## 9.5 IDCHAR Function.

The IDCHAR function takes three arguments, one real array A and two integer variables N and I. It returns an integer value.

The array A of dimension N should be filled with left justified characters, one character per word. (Most often A is dimensioned with N=80 and consists of a card image read under 80A1 Format in INPUT). The function IDCHAR then examines the Ith element of the array, that is A(I).

The function IDCHAR is then assigned a value as follows:

IDCHAR = 0 if A(I) is a number or minus sign (hyphen);
IDCHAR = 1 if A(I) is a letter or an apostrophe;
IDCHAR = 2 if A(I) is a decimal point (period);
IDCHAR = 3 if A(I) is a blank;
IDCHAR = 4 if A(I) is a separator (i.e., a comma, a left or right parenthesis, a slash, or a question mark);
IDCHAR = 5 if A(I) is a dollar sign; and
IDCHAR = 6 if A(I) is anything else.

### 9.6 NUMRIC Function (Logical Valued).

The NUMRIC function takes one real array argument, B, and three integer variable arguments N1, N2, and IB.

The array B of dimension N1 by N2 (i.e., DIMENSION B(N1, N2)) is checked to determine whether its IBth line (that is, entries B(J, IB) for J = 1, 2, 3,..., N2) is numeric data (i.e., whether it consists only of numerals, minus sign, and decimal point possibly with trailing blanks).

If the line is numeric data, NUMRIC is assigned the value TRUE. If the line contains some non-numeric characters (letters, commas, etc.), then NUMRIC is assigned the value FALSE.

The IBth line of the B array is treated as a left justified character string (10 characters per word) which is examined character by character. Only the first 80 characters of the line are examined.

When the line being checked meets certain special conditions, other actions may be taken: If more than one decimal point is encountered, an error stop with message occurs. If no decimal point is encountered but the characters are otherwise numeric data, then a decimal point is added to B at the end of the non-blank characters, NUMRIC is assigned the value TRUE, and a return occurs. If the data are numeric but consist of more than ten non-blank characters (including the added decimal point, if any), then an error stop with message occurs. If the data contain imbedded blanks, then checking stops at the first imbedded (i.e., non-trailing) blank and the program behaves as though every character following were a blank. (This should have no effect in practice since the only imbedded blanks occur in PEDATA record names entered with the $ preceding them. Those names will already have failed the numeric test before ever reaching their first imbedded blanks).

### 9.7 URAN31 Function.

This function takes one integer argument I and returns a real value between 0 and 1 (exclusive).

This is the "random" number generator used to sample most COPE data and to generate "random" numbers to be compared against probabilities to determine whether various tests are passed.

The best results are obtained when I is an odd integer in the range from 0 to 67108864. URAN31 generates a sequence of "random" numbers by replacing I by 3125*I(MOD 67108864) and then setting URAN31 equal to I divided by 67108864. (That is, a linear congruential type random number generator A(N+1) = A(N)*B(MOD C) where B = 3125 and C = 2**26). Then repeating the process with the new value of I gives the next "random" number.

The function is written so that overflow will not occur even on machines with a word size as small as 32 bits.

Normally, one would not define the random number generator explicitly in the program code since one is available on nearly every computer as a standard system function. However, because users on other machines may wish to run the sample cases (see appendices) to check for correct implementation, the URAN31 function has been built into COPE as its standard (system independent) random number generator.

### 9.8 ABRTTL Subroutine.

This subroutine takes the integer argument IABRT which is the number of the test whose failure caused ABRTTL to be called.

ABRTTL updates the test abort total array NABORT by increasing the proper entry of the NABORT array by one. For mission aborts, the elements corresponding to all rounds of the IABRTth test are increased; for round aborts, only the element corresponding to the IABRTth test and current (KRFth) round is increased.

In addition, the flag ENDREP is set to TRUE to indicate that a test has been failed and the current replication (or round) need not continue through the subsequent tests.

### 9.9 ADDTBR Subroutine.

This subroutine adds TBR (time between rounds) to TIMRA (time round arrived) to obtain the new value for TIMRA (time next round arrives).

### 9.10 ADDTOF Subroutine.

This subroutine adds TOF (time of flight) to TIMNOW (time gun is ready to fire Copperhead) to obtain the TIMRA (time first round arrives).

### 9.11 ARTCHK Subroutine.

This subroutine performs the test to determine whether the FO (or DO, designator operator) has been killed or suppressed by RED preparatory artillery fires on this replication.

## 9.12 BLOCHK Subroutine.

This subroutine performs the bail out checks (tests 7 and 10).

## 9.13 CCPLOT Subroutine.

This subroutine would produce a Cal-Comp plot of case results, if implemented. At the present time it merely produces a comment that no plot was produced for the case.

## 9.14 COMMNT Subroutine.

This subroutine takes four arguments: one real array A containing the character string that is the comment, one integer N1 which is the dimension of A, and two logical flags that indicate whether the comment is a user comment and whether the comment is a program generated "$$$" comment.

The subroutine writes the comment immediately (along with a "$$$" comment page heading, if needed) if the comment is a "$$$" comment (either user or program generated). If the comment is a "$$" comment, then it is encoded into the CMENT array for later printing with the current case. If the comment is a "$" comment, it is ignored and a return occurs.

Note that user comments are in format 80A1 with DIMENSION A (80) whereas program generated comments are in format 8A10 with DIMENSION A(10). This difference need not concern the casual user, but is required knowledge if one is to modify the program in so far as comments are concerned.

## 9.15 CREAD Subroutine.

This subroutine is called by INPUT whenever the user exercises the temporary option. The second field on the TEMPORARY card will be used by subroutine INPUT to set the value of IT2 equal to the keyword number for which the temporary option is being used.

CREAD then sets ITYPE equal to IT2-20 and executes a computed GO TO on ITYPE. As a result of this computed GO TO, control transfers to one of eight sections where the formatted temporary data is read. Each of these sections corresponds to one of the eight data blocks and each consists mainly of READ statements that fill in that data block's variables. When the reading is completed, control branches to statement number 410 where a return occurs.

## 9.16 DETCTN Subroutine.

This subroutine samples the detection time distribution to obtain the detection delay time for this replication (i.e., time from unmask until D.O. initiates call for fire. It includes detection, acquisition, and decision time).

If "random occurrence" is being played, control returns. If "shooting gallery" is being played, the detection delay time is compared with the line of sight duration (test 4). If the test is passed, a return occurs; if the test is failed the abort totaller is called (ABRTTL) prior to the return.

9.17 DFAULT Subroutine.

This routine merely sets the default values for many of the case descriptor parameters, data block record names, and flags. (It could really be replaced by some data statements to fill in these same variables if one so desires).

9.18 DFCHK Subroutine.

This subroutine performs test 14. If the FO (DO) has been killed on a previous call for the current replication, control transfers to 120 where the round abort is totalled (call to ABRTTL) followed by a return.

Otherwise, the routine calculates the current designator-to-target range. Then it uses that range value to interpolate in the DFFOKL array to obtain probability that direct fire kills the DO and probability that direct fire obscures but does not kill the DO. When these probabilities are obtained, test 14 is performed to determine whether the DO is unaffected, obscured but not killed, or killed. If the DO is unaffected, the routine returns; if the DO is obscured but not killed, the routine calls ABRTTL to increment the round abort counter and then returns; and, if the DO is killed, the routine increments the DO killed counter, sets the DO killed flag, then calls ABRTTL and returns.

9.19 DUST Subroutine.

This subroutine performs test 6 and calls ABRTTL if the test is failed.

9.20 ECHO Subroutine.

This subroutine takes the integer argument IUNIT. IUNIT is the logical I/o unit to which ECHO will write. (IUNIT = 8 currently)

Subroutine ECHO does nothing more than write out the current values of the parameters, flags, arrays, etc. that describe the conditions of the case. (see section 7.6) This printout is provided primarily as a debugging aid.

9.21 FINDIT Subroutine.

This subroutine takes a real array B of character strings dimensioned N1 by N2 and returns an integer valued array IT dimensioned N3. Normally N1 = 8, N2 = 10, and N3 = 10 so we have

DIMENSION B(8, 10), IT (10). The Kth line of B (i.e., B(J,K) for J = 1,2,3,...,8 with K fixed) consists of the character string found in the Kth field of the input card just read. This character string is compressed by eliminating blanks (the exception is that blanks occurring in PEDATA record names preceded by $ are preserved when moved into the B array) and packing the results 10 characters per word into B.

Subroutine FINDIT compares the first line of the B array (B(1,1) through B(8,1)) with the keywords in the CHAR array in an attempt to find a match. If not successful, an error message and error stop occur; if a match is found, then IT(1) is set to the keyword number (contained in CHAR array also).

Next, if the match occurred with the TEMPORARY, OVERRIDE, or RESET keywords, then a second pass through the keyword list (CHAR array) is made to find a match with the second line of the B array. If this search is successful, IT(2) is set to the value of the matched keyword and a return occurs. If unsuccessful, an error message and stop occurs. An error message and stop also occur if a match is found for a keyword that is not allowed with TEMPORARY, OVERRIDE, or RESET.

If the first match was not with TEMPORARY, OVERRIDE, or RESET, then the routines checks CHAR (2, IMATCH) to see how many option word character strings are allowed with the IT(1)th keyword. If this number is zero, that is the flag to the program that it may have one of the keywords that takes only numerical values and so control transfers to statement 210. At statement 210, B(1,2) is checked. If B(1,2) is blank, a return occurs; if B(1,2) is numeric data, IT(2) is set to zero; if B(1,2) is neither blank nor numeric an error print occurs. Note that the blank occurs either when the keyword has no optionwords allowed (as with <u>FIRST CASE</u>, <u>END</u>, etc.) or when the keyword is omitted (i.e., the default valued is desired).

If CHAR(2, IMATCH) is non-zero, it gives the number of different option word character strings allowed with the current card's keyword. The TAPE 11 record name is then set to the abbreviated form of the keyword and the record of option word character strings allowed with the keyword is read into the CHARA array. The program then proceeds to match B(2,1) through B(2,8) with allowable option word choices for option word 1, B(3,1) through B(3,8) with allowable choices for option word 2, etc. If a match is not found for any line of B, then it is tested to see whether it is numeric; if not, an error message and stop occur, if it is numeric, the next line of B is checked. If line K of B is numeric, IT(K) is set to zero; if line K of B is blank, IT(K) is left at one (the value indicating default) to which it was set near the beginning of FINDIT; if line K of B matches one of the character strings allowed for option word K-1, then IT(K) is set equal to the number of the option word choice.

It is through setting the IT array that FINDIT signals subroutine INPUT as to which keyword and optionwords were found on the input card. Of course, in the case of numeric data or OVERRIDE record names, the values in the B array will have further use even after IT has been determined.

9.22 GETRNG Subroutine.

This subroutine samples the cumulative distribution of acquisition ranges (actually designator-to-target ranges at time of unmask) to obtain the initial designator-to-target range for the current replication. The cumulative distribution of line-of-sight segment lengths corresponding to the range bracket in which the designator-to-target range lies is then sampled to obtain the length of the segment of the target's path that is visible to the DO. This LOS segment length is then added to the target column's length and the sum is divided by the target velocity to obtain the duration of line of sight (i.e., the length of the time interval from target unmask until LOS to last vehicle in column is lost). Next a return is executed.

If the target velocity is zero (or negative) an error stop with message occurs.

9.23 GETTIM Subroutine.

This subroutine serves three basic functions: (1) perform test 8, (2) perform test 9, and (3) calculate the response time (time from DO call for fire until gun is ready to fire Copperhead) for those missions that pass tests 8 and 9.

As mentioned in Chapter 2, there are three different ways of playing communication and processing delay times.

(1) If parameterized response time (PRSPTM) is non-zero, then this routine sets the time the gun is ready to fire (TIMNOW) equal to PRSPTM. It then performs test 8 and 9. If either test fails, ABRTTL is called and a return occurs; otherwise, ABRTTL is not called and the routine executes a return with TIMNOW = PRSPTM.

(2) If parameterized response time is zero, then the routine is to sample to obtain response time. If the variable FTSILL is TRUE, the routine samples the single response time distribution (FSRT array) to obtain the response time which is then added to the detection time to obtain the time at which the gun is ready to fire. It then performs tests 8 and 9. If either test fails, ABRTTL is called and a return occurs; otherwise, ABRTTL is not called and the routine executes a return with TIMNOW set at the value to be used for this replication.

(3) Finally, if parameterized response time is zero and the variable FTSILL is FALSE, then the routine uses the sum of the separate response time components as the response time. If digital communication is modeled, then the time to enter digital data into DMD is sampled and added to current time (TIMNOW which was previously set to DETTIM).

Then an unnumbered test is made to determine whether digital commo is successful. If it is, a transmission time is added to TIMNOW and control transfers to statement 160 (see below). If digital commo fails, then a test is performed to determine whether voice commo is successful. If voice is successful, then a time to switch from digital to voice and the voice commo transmission time are added to TIMNOW before control transfers to statement 160. If voice is also unsuccessful, then test 8 is considered to have been failed, ABRTTL is called, and a return executed.

If digital commo is not selected, then voice commo only is played. In this case, the test for successful voice commo is made (test 8) and if failed, an abort (call to ABRTTL) occurs followed by a return. If the test is passed, a voice commo time is added to TIMNOW and control reaches statement 160.

At statement 160, the battery computer system processing and battery response time are added to TIMNOW. Test 9 is then performed. If it fails, ABRTTL is called and a return executed; if it is passed, then TIMNOW is returned to be used for the current replication.

## 9.24 GETVIS Subroutine.

This subroutine samples the weather data used in the current case to obtain the cloud ceiling altitude and the visibility range limit.

The routine first randomly determines whether there is a cloud ceiling. Next it randomly determines whether there is a cloud free line-of-sight for the COPPERHEAD trajectory.

If there is a cloud-free line-of-sight, then the cloud ceiling altitude is set to its highest value and a visibility limit is determined by random sampling using the PRGCFL array. Control then branches to statement 160 (see below).

If there is no cloud-free line-of-sight, then the cloud ceiling altitude and visibility limit are both determined by a random sampling from the W array.

Note that the items of importance that are defined by this routine are the subscript of the cloud ceiling altitude (ICC), the subscript of the visibility range limit (IVL), and the visibility range limit itself (VISLIM). Also, note that ICC is initially a function of increasing altitude (that is, higher altitudes correspond to larger values of ICC); however, because the ordering convention for cloud ceilings was reversed between the weather data and the PE data, ICC is redefined after statement

160 by setting the new value of ICC equal to seven minus the old ICC value (so that higher altitudes correspond to lower values of ICC).

9.25 HEADER Subroutine.

This subroutine produces the heading page for each case.

The heading page is produced by encoding the lines for the first column of boxes into the AOUT array; those for the second column into the BOUT array; and those for the third column into the COUT array.

Next the routine generates some comments regarding time inconsistencies, record name overrides, and data read by the TEMPORARY option as applicable.

Finally, following the lines giving case number and number of replications (which are written before HEADER does any encoding) HEADER writes the AOUT, BOUT, and COUT arrays to produce the nine boxes of case description information that occur on the case header page. Then if any "$$" comments are to be printed for this case, the CMENT array is written (this includes both user comments and program generated comments).

9.26 HITCHK Subroutine.

This subroutine performs test 18.

First, it interpolates with respect to designator-to-target range (RNGNOW) in the proper row of the TTF array as determined by designator type and target posture (exposure) to obtain the probability of hit. This probability of hit is really the probability of hit given that the round engages (that is, that the seeker locks onto the laser spot on the target and does so within the maneuver footprint of the COPPERHEAD round).

Next the routine compares a random number (between 0 and 1) to the probability of hit to determine whether the current round hits. If the round hits, a return occurs; if not, ABRTTL is called to record the failure (miss) and a return occurs.

9.27 INITLZ Subroutine.

The subroutine initializes certain case independent values.

After setting IRAY and calling SYSTEMC to set the non-standard error recovery process, the routine proceeds to set the values of CHAR (2,I) for I = 1,2,...NREC from the values in the record named OPTNNUMS on Tape 11. (This record is automatically updated by the PREPMS program each time new optionword character strings are added for a keyword).

The program next calls DFAULT to set the default values for the various case descriptors, record names, etc. Then USET is called to set the values to be used as the base to which the first case input card values are to be applied. Finally, MSABLM is set and a return is executed.

## 9.28 INPUT Subroutine.

This subroutine reads the input lines for a given case and sets the variables in the program to the values required to run the case.

This routine is the longest and one of the most complex in the COPE model. It is most easily followed by means of a flowchart (see Chapter 11); however, an attempt to sketch its main sections is given here.

(1) First, the routine checks to determine whether there is another case to read. If not, it stops. If there is more to read, the routine first sets IRAY and calls SYSTEMC to establish non-standard error recovery. Then it clears various arrays and counters and increments the case number.

(2) Next it reads input cards until it encounters an END, ENDF, or end-of-file. For each card that it reads, it first checks whether that card is a comment card; if it is, subroutine COMMNT is called and the next card is read. If the card is not a comment, the routine calls SEPREC which separates the fields on the input card and moves each of them into a line of the B array. Then subroutine FINDIT is called to match each line in the B array against the keywords and option words allowed in that field. This matching determines the values of the IT array.

The program next checks to see whether the current card supersedes some of the previous cards for the current case. If it does, a comment is generated; in either case, the routine next branches to one of five statements depending on whether the card is a FIRST CASE, NEXT CASE, END, ENDF, or any other card, respectively. If the card is an END or ENDF, control transfers to statement 490 (see below). If the card is a NEXT CASE, RD (or equivalent), USET is called to establish the default values as the base to which changes caused by the input cards are to be applied; if the card is NEXT CASE, DR, then USET is not called and the changes caused by the input cards are applied to the conditions in effect for the previous case. Either type of NEXT CASE (or FIRST CASE) causes control to then read the next card. If the card is not an END, ENDF, NEXTCASE, or FIRST CASE, then control reaches statement 200.

At statement 200, a many branched computed GO TO is executed; the branch followed depends on the keyword number (IT1 or IT(1)). The actions taken for each case of the computed GO TO are varied but the main activities are to fill the UVALUE array with either numerical values, record names, or flag values. If the keyword was TEMPORARY, CREAD is called; if OVERRIDE, the record name is set; and if RESET, the default for that keyword is reactivated. Each case of the computed GO TO ends with a GO TO returning control to the line that reads the next input card.

(3) At line 490 the routine starts filling in the values of the various parameters that describe the current case (flags, arrays, single variables, etc.). These are generally filled in by calls to READMS to read records to fill in data blocks (record names as established when filling in UVALUE array), by equating variables to UVALUE numerical values, by looking up values in pre-defined arrays according to a subscript value in UVALUE array, by calling a special subroutine (in the case of PEDATA) to generate a record name and then reading that record from TAPE 11 into the data block, or by calling special routines (such as PSMOKE) that use inputs to calculate a parameter value. In any event, by the end of INPUT all variables required to run the case have been set and the replications can begin.

(4) INPUT finishes by calling HEADER and ECHO and then returning.

## 9.29 LOSCHK Subroutine.

This subroutine is used to determine whether the DO still has (macro-terrain) LOS to the target when the round arrives (test 12).

It performs the test in two different ways depending on the LOS model used.

For "shooting gallery", it merely compares the time the round arrives with the duration of the line-of-sight. If the round arrives before the line-of-sight ends, a return is executed; if the LOS ends before the round arrives, ABRTTL is called prior to the return.

For the "random occurrence" LOS model, the probability P of LOS to a single vehicle in the current designator-to-target range bracket is found. The probability that at least one vehicle is in view is then taken to be equal to 1-(1-P)**XNV where XNV is the number of target vehicles currently unkilled. Once this probability is obtained, it is compared to a random number and the result settles test 12. If test 12 is passed, a return occurs; if it is false, ABRTTL is called,then a return occurs.

### 9.30 MINTRN Subroutine.

This subroutine performs test 16 and sets target posture (exposure) with respect to mini-terrain.

First the current designator-to-target range is computed. Then this range is used to interpolate in each of the three rows of the PSTTBL array. As a result of this interpolation the PSTVAL array is filled with the probabilities of completely obscured target posture, fully exposed target posture, and hull defilade (turret exposed) target posture respectively.

A random number is then drawn and compared to these probabilities to determine which of the three target postures occurs. If the target posture is "completely obscured", then test 16 is failed, ABRTTL is called, and a return occurs; otherwise, ITGTPS is set to indicate the target posture (1 for fully exposed; 2 for hull defilade) to be used in hit and kill probability tests and a return occurs.

### 9.31 NOREC Subroutine.

This routine is called indirectly through the CDC non-standard error recovery facility (SYSTEMC routine explained in CDC Fortran 4 Extended) It is called whenever READMS attempts to read a record from TAPE 11 using a record name that does not correspond to any of the records on TAPE 11.

The routine writes a message giving the record name that could not be found, executes a subroutine call trace and then executes a stop with message.

### 9.32 OUTPUT Subroutine.

This subroutine writes the results page for each case.

The routine begins by clearing several arrays.

Next, it calculates the percentage of total missions that have a failure (abort) at each test (for "round" aborts, this percentage is calculated for each round). This is the ABRPCT array.

Then it calculates what percentage of the missions that reached a given test passed it. Again this is calculated for each round for the "round" aborts. This is the PASPCT array.

Next the program sets the number of occasions, attempted engagements, and shots.

Then it compares the number of kills as calculated by two different counters and, if there is a disagreement in value between them, it prints a message (see Section 7.3.1).

Next, it calculates the probability of kill given an occasion, the probability of kill given an attempted engagement, and the probability of kill given a shot. These are calculated separately for each round of the mission.

Then it calculates the probability of a shot given an occasion, probability of a shot given an attempted engagement, and probability of attempted engagement given a shot. It also calculates the probability of having line-of-sight on each round.

Next it compares the number of missions (or rounds, as appropriate) tested for each abort condition as computed in two different ways. If the results differ an error message is generated (see section 7.3.2).

Then the program calculates the FO (DO) survivability measures; that is, probability DO is killed and probability DO is not killed given that the mission reached that test.

Now that all of the values to be printed on the results page have been calculated the routine proceeds to print the results in the order they appear on the case results page (see Chapter 6).

Note that prior to printing percentages, function FPRCNT is applied to format the percentages for output. Also note that any percentage valued variables with the value -1.0 are printed as "UNDEF" indicating undefined value (this occurs when the percentage could not be calculated due to a zero denominator. It is for this reason that all percentage valued variables in OUTPUT are initialized at -1.0 and then only assigned values if the ratio used in computing the percentage has non-zero denominator).

### 9.33 PECHK Subroutine.

This subroutine performs test 17 (i.e., does COPPERHEAD round engage the target?)

First it calculates the delay time (DELTAT) for the current round. By delay time is meant the time the round arrives minus the expected arrival time (which is based on nominal response time). Next the value of IMUTS (the "MUTS" factor -- see Section 16.1) is determined and the range value indices to be used for interpolation are set.

If IMUTS is 2, it means that not all vehicles in the target column have passed the point of closest approach to the centroid of the COPPERHEAD footprint. That is, delay has no effect and so PE is computed by interpolating with respect to designator-target range alone in the PETBL array (using delay time equal to zero (IDELTT=1) and holding cloud ceiling and visibility limit constant at their previously sampled values (obtained in GETVIS)).

If IMUTS is 1, it means that all vehicles in the target column have passed the point of closest approach to the centroid of the COPPERHEAD footprint. That is, delay time has an effect and so PE is computed by interpolating with respect to both designator-target range and delay time while holding cloud ceiling and visibility limit constant.

In either case, once PE (probability that seeker engages target and target is within Copperhead maneuver footprint) is computed, it is compared to a random number sampled from a uniform distribution (from 0 to 1). If the random number is less than PE, a return occurs; otherwise, ABRTTL is called before the return occurs.

### 9.34 PENAME Subroutine and PEIDNT Entry.

This subroutine takes ten arguments. When the call is to PENAME the first nine are "input" and the last one is "result"; whereas a call to PEIDNT uses the current value of the tenth argument as "input" and returns the values for the other nine.

This subroutine is used to create PE data record names. The values of the nine parameters (see section 5.4) are taken as arguments (in the same order they are described in section 5.4). They are loaded into the AVALUE array and are searched for in the XVALUE array. When matches are found, the corresponding XVALUE entries (for XVALUE (I,J,2)) are used to construct the record name which is returned as IDCODE.

When PEIDNT is called, the record name is decomposed into the separate XVALUE (I,J,2) values (and TR value) used to create it. The corresponding XVALUE (I,J,1) values are found and equated to the nine parameters (note slightly different handling of TR, nominal response time). These values are then returned.

Note that while every choice of the nine parameters can give at most one record name, there may be more than one choice of parameters that can give that record name. For example, if all parameters are held constant except target velocity (VEL), then VEL = 2 m/s and VEL = 3 m/s will give the same record name (using the current XVALUE array). Hence, when that record name is decoded by PEIDNT to obtain the parameters to which it corresponds, it will have no way of choosing between 2 m/s and 3 m/s for VEL; instead it will merely take the first one that matches (2 m/s in this case).

### 9.35 PKCHK Subroutine.

This subroutine performs test 19 (i.e., did the COPPERHEAD kill the target?)

The routine looks up the PK (probability of kill given a hit) in an array as a function of target type and target posture (exposure). This value is then compared to a random number sampled from

a uniform distribution (from 0 to 1). If the random number is less than PK, then a kill occurs, the number of vehicles killed on the KRFth round for this case (RNDREC) is incremented, the number of vehicles killed (NVEHKL) on this replication is incremented and a return occurs. If not, ABRTTL is called and then the return occurs.

### 9.36 PPLOT Subroutine.

This subroutine is currently just a "stub" that producess a "$$" program generated comment saying that no printer plot was produced for the current case.

If implemented, this routine would produce printer plots (bar graphs) giving a graphical presentation of case results.

### 9.37 PSMOKE Subroutine.

This routine calculates the probability that smoke aborts the potential COPPERHEAD mission. This probability is a function of the number of each of two types of smoke rounds fired (NSMK2 and NSMK5) and the distribution of Pasquill atmospheric stability categories, distribution of wind speed velocities, and distribution of relative humidity levels corresponding to the current case's weather (i.e., month and time of day).

The entire routine reduces to evaluating PSMKKL (probability smoke kills mission) according to the formula:

$$\mathrm{PSMKLL} = \frac{1}{\mathrm{FRONT}} \sum_{I=1}^{3} \sum_{J=1}^{3} \sum_{K=1}^{2} \left\{ \mathrm{MIN}\left( \frac{\mathrm{NSMK2}}{\mathrm{SMK2(I,J,K)}} + \frac{\mathrm{NSMK5}}{\mathrm{SMK5(I,J,K)}}, \mathrm{FRONT} \right) * \mathrm{WNDSPD(I,J)} * \mathrm{HUMID(K)} * \left[ \mathrm{PASQL(2*I-1)} + \mathrm{PASQL(2*I)} \right] \right\}$$

where all of the variables are as defined in the Glossary of Variables (Chapter 12).

Note that because certain combinations of windspeed and Pasquill category never occur, the SMK2(I,J,K) and SMK5(I,J,K) values corresponding to them have been left blank (i.e., equated to zero). To avoid the division by zero that would otherwise result, the range of J in the summation is sometimes limited in the routine to avoid certain SMK2 and SMK5 values which should have no weight in the sum anyway.

### 9.38 REINTZ Subroutine.

This subroutine resets certain counter arrays to zero. It also resets the random number seeds for each case and on the first case it writes the seeds on Tape 8.

Note that the seeds are set according to the formula:

IR(I) = 2*(MOD(8**I, 29) *2**25/29) + 1

for I = 1,2,..., 28.

This assures that each IR is odd and that the IR's are approximately uniformly spaced between 0 and $2^{26}$.

9.39 RNDREL Subroutine.

This subroutine performs the test of round reliability (test 15).

9.40 RNGCHK Subroutine.

This subroutine performs test 3.

It compares designator-to-target range at unmask to the maximum designator range. If the maximum designator range is less than the designator-to-target range, ABRTTL is called and a return occurs; otherwise, the test is passed and a return occurs.

9.41 SEPREC Subroutine.

This subroutine takes a real array A of dimension N1 and breaks it up into fields (i.e., separates records) and compresses them one per line into the B array which is of dimension N2 by N3.

The routine first clears several arrays to be used in separating the fields on A. Then it calls IDCHAR once for each element of A to determine what kind of character it is. A is assumed to be an array (usually DIMENSION A(80)) of characters filled by reading one input card under 80A1 format.

If the routine encounters sequence numbers in columns 73-78 of the first card (i.e., elements 73-78 of array A) and blanks in 63-72 and 79-80, then it issues a "$$$" comment (see section 7.1) and removes the sequence numbers on the first card and all subsequent cards (A arrays) of the current run. If it encounters something other than the blanks and sequence numbers as described above in columns 63-80 of the first card, then it leaves columns 73-78 unchanged on all cards of the current run.

Next the program separates the fields on the card (i.e., in A) and compresses them into B. It does this by copying one character at a time from A to PB omitting blanks (except when part of a ten

character string preceded by a "$" sign) and starting a new line of PB every time a separator (comma, left or right parenthesis, question mark, or slash) is encountered. It then encodes each line of PB into a line of B so that each element of B contains up to ten characters.

This means that B(I,J) for I = 1,2,...,8 contains the characters of the Jth field of the input card compressed by removing blanks (with the exception noted) and packed ten characters per word (until the characters run out).

A return now occurs with the B array filled with the character strings to be compared to the keywords and option words allowed.

Note that an input card cannot have more than ten fields nor fewer than one. A field on an input card (or in A) is defined as a string of characters lying between two successive separators where for this definition we include the beginning of the card and the end of the card as separators and do not count characters that would otherwise be separators if they lie in a ten character string following a dollar sign. (The normal separators are mentioned above in this section).

### 9.42 SMOKE Subroutine.

It compares the value of PSMKKL calculated in PSMOKE with a random number sampled from a uniform distribution to determine whether smoke was sufficient to abort the potential Copperhead mission. As usual, if an abort occurs, ABRTTL is called prior to returning; otherwise, an immediate return occurs.

### 9.43 SMPLCD Subroutine.

This subroutine randomly samples from a cummulative distribution. As arguments, it takes an array A, its dimensions ND1, ND2, the row number NRS containing the values to be sampled, the number of entries NDR used in the row ($NDR \le ND1$), and the random number seed IRNDM. It returns the value sampled XSMPL and the class ICLASS in which it falls.

This routine requires an array A with A(I,1) for I = 1,2,., $NDR \le ND1$ consisting of probabilities $A(1,1) = 0.0 \le A(2,1) \le \dots A(NDR,1) = 1.0$. The routine samples a random number RN from a uniform distribution (from 0 to 1) and then brackets this number with two entries from A(I,1) such that $A(I,1) \le RN \le A(I+1,1)$.

The corresponding entries A(I,NRS) and A(I+1,NRS) from the NRSth row of A are obtained. The value sampled XSMPL is then defined by straight line interpolation, (i.e., XSMPL is defined so that the point (RN, XSMPL) lies on the line segment joining (A(I,1), A(I,NRS)) to (A(I+1,1), A(I+1,NRS))).

Put in terms of a formula, that means:

$$XSMPL = A(I,NRS) + (RN-A(I,1))*(A(I+1,NRS)-A(I,NRS))/(A(I+1,1) -A(I,1)).$$

ICLASS is then defined to be equal to I and a return occurs.

Note that A defines a cummulative distribution. That is, when sampling.

$$Pr(XSMPL \leq A(I,NRS)) = A(I,1) \text{ for } I = 1,2,\ldots,NDR.$$

Hence, the user must take care to use only cummulative distributions for the input data that will eventually be sampled.

9.44 TIMCHK Subroutine.

This subroutine performs test 11 (i.e., does the DO still have line-of-sight to the target when the battery is ready to fire?)

It is called only for the "shooting gallery" LOS model.

9.45 TITLE Subroutine.

This routine merely writes the Title page of the output (see Chapter 6). It also sets the date and time that the current run began.

9.46 USET Subroutine.

This subroutine is called when the next case to be run is to use the baseline (default) values of the case parameters as the base to which its own input cards are to be applied.

The routine sets the UVALUE common block and a few other variables to their default values. Whatever inputs the next case then contains are applied to changing these variables from their defaults rather than from the previous cases values.

9.47 VISCHK Subroutine.

This subroutine performs test 2.

It compares designator-to-target range at unmask to the visibility range limit. If the visibility range limit is less than the designator-to-target range, ABRTTL is called and a return occurs; otherwise, the test is passed and a return occurs.

9.48 WARNDO Subroutine.

This subroutine performs test 13.

If the current round is the first of this COPPERHEAD mission, then a random number is sampled from a uniform distribution and compared to the probability that DO is warned to lase to determine whether the DO received the message to begin lasing. If he did receive the message, the test is passed and a return occurs; if not, ABRTTL is called prior to returning.

If the current round is not the first round of this COPPERHEAD fire mission, then DO is assumed to have been warned either by communication or by sighting the impact of the first round and, in either case, turns on his laser so test 13 is passed and a return occurs.



Next page is blank.

CHAPTER 10

# 10. MEMORY ORGANIZATION: COMMON BLOCKS AND EQUIVALENCE CLASSES

## 10.1 Common Blocks.

This chapter describes the common blocks and equivalence classes in COPE. This organization of the values stored in memory is a matter of some importance to anyone attempting to modify the program.

Tables 10-1 and 10-2 give, respectively, a list of the common blocks in each subprogram of COPE and a list of the subroutines in which each common block occurs.

## 10.2 Equivalence Classes.

Table 10-3 gives the equivalence classes. There are twelve equivalence classes used in COPE (some are used in more than one subroutine). The equivalence classes are described below in alphabetical order of the names of their first items. Eight of the twelve equivalence classes are the eight data blocks referred in Section 4.5.1.:

ACQDAT is the Acquisition Range data block,
AINVDA is the Invariant data block,
DFSDAT is the Direct Fire Suppression data block,
PEDATA is the Probability of Engagement data block,
STDAT is the Target Posture Distribution data block,
RODATA is the Random Occurrence data block,
RSPDAT is the Response Time data block, and
WDATA is the Weather data block.

These data block equivalence classes (along with a MAX equivalence class which has nothing to do with COPE's use of PREPMS) are also present in the PREPMS program.

Each equivalence class is formed by an equivalence statement establishing that one array or non-subscripted variable occupies the same memory locations as a set of one or more other arrays or non-subscripted variables. In each case the one array or non-subscripted variable (after which the equivalence class is named in Table 10-3) occupies all of the computer memory used by that equivalence class and so is dimensioned with a value equal to the entire length of the equivalence class (if the length is 1, then the equivalence class occupies but one word of computer memory and the class is named after a non-subscripted variable).

In Table 10-3, the information about each equivalence class is separated by dashed lines and is read as follows:

The first entry on the left gives the name of the equivalence class.

The next entry (length) gives the length of the equivalence class in terms of the words of computer memory occupied by the members of the equivalence class. This length is also the dimension of the array (or non-subscripted variable if length is one) in the first entry.

The entries under "MEMBERS" give bias, name, and length of each member. That is, for each member (variable) of the equivalence class (other than the member in the first entry above) there are three items given: first the bias or number of words preceding the variable in that equivalence class in computer memory space, second the variable (or array) name, and third the length of the array (or non-subscripted variable) in terms of computer words (one for non-subscripted variables).

Note that for each equivalence class, the memory space occupied is part (or all) of some common block. This correspondence is as follows:

ACQDAT is in the RNGLOS common block,
AINVDA is in the HIT common block,
D is in the DVALUE common block,
DFSDAT is in the DODF common block,
ICODE is in the ICODE common block,
PEDATA is in the HIT common block,
PSTDAT is in the HIT common block,
RECODE is in the RECNAM common block,
RODATA is in the RANDOC common block,
RSPDAT is in the RSPTIM common block,
U is in the UVALUE common block, and
WDATA is in the WEATHR common block.

Also, note that the variables in an equivalence class may be used (under their names in the COMMON statements) in subprograms other than those in which the equivalence class is defined.

TABLE 10-1 COMMON BLOCKS IN EACH SUBPROGRAM OF THE COPE PROGRAM

| SUBPROGRAM | COMMON BLOCKS | | | | | | |
|---|---|---|---|---|---|---|---|
| ABRTTL | ABORT | FLAG | RPLCTN | | | | |
| ADDTBR | TIME | | | | | | |
| ADDTOF | TIME | | | | | | |
| ARTCHK | KTEST | MISC | RANDOM | RPLCTN | | | |
| BDATA1 | ABRLBL | ACHAR | DESRNG | DISPLY | FLTTIM | FSRESP | HEADNG |
| | LOGFLG | POINT | SMOKED | STITLE | SYMBOL | XVALUE | |
| BLOCHK | BAIL | KTEST | RANGE | RPLCTN | TARGET | TIME | |
| CCPLOT | RUNDAT | | | | | | |
| COMMNT | COMENT | | | | | | |
| COPE | ABRLBL | ACHAR | BAIL | DESRNG | DISPLY | DODF | FLAG |
| | FLTTIM | FSRESP | HEADNG | LOGFLG | POINT | RPLCTN | RUNDAT |
| | SMOKED | STITLE | SYMBOL | TIME | XVALUE | | |
| CREAD | DODF | HIT | RANDOC | RNGLOS | RSPTIM | WEATHR | |
| DETCTN | FLAG | KTEST | RANDOM | RPLCTN | RSPTIM | TIME | |
| DFAULT | DVALUE | RUNDAT | | | | | |
| DFCHK | DODF | KTEST | RANDOM | RANGE | RPLCTN | TARGET | TIME |
| | WEATHR | | | | | | |
| DUST | KTEST | MISC | RANDOM | RPLCTN | | | |
| ECHO | BAIL | DODF | FLAG | HIT | LOGFLG | MISC | OVER |
| | PEDESC | RANDOC | RANGE | RECNAM | RNGLOS | RPLCTN | RSPTIM |
| | RUNDAT | SMOKED | TARGET | TIME | WEATHR | | |
| FINDIT | ACHAR | ICODE | INDEX1 | RUNDAT | SYMBOL | | |
| GETRNG | RANDOM | RANGE | RNGLOS | RPLCTN | TARGET | TIME | WEATHR |
| GETTIM | FLAG | FSRESP | KTEST | LOGFLG | RANDOM | RPLCTN | RSPTIM |
| | TIME | | | | | | |
| GETVIS | RANDOM | RANGE | RPLCTN | WEATHR | | | |
| HEADER | BAIL | COMENT | DISPLY | FLAG | HEADNG | LOGFLG | MISC |
| | OVER | PEDESC | RANGE | RECNAM | RPLCTN | RSPTIM | RUNDAT |
| | SMOKED | SYMBOL | TARGET | TIME | UVALUE | | |
| HITCHK | HIT | KTEST | RANDOM | RANGE | RPLCTN | RSPTIM | TARGET |
| | TIME | | | | | | |
| IDCHAR | SYMBOL | | | | | | |

TABLE 10-1 COMMON BLOCKS IN EACH SUBPROGRAM OF THE COPE PROGRAM - CONTINUED

| SUBPROGRAM | COMMON BLOCKS | | | | | | |
|---|---|---|---|---|---|---|---|
| INITLZ | ABORT | ACHAR | DVALUE | FLAG | ICODE | INDEX1 | RUNDAT |
| | UVALUE | | | | | | |
| INPUT | ABORT | BAIL | COMENT | DESRNG | DISPLY | DODF | DVALUE |
| | FLAG | FLTTIM | HIT | ICODE | INDEX1 | LOGFLG | MISC |
| | OVER | PEDESC | POINT | RANDOC | RANDOM | RANGE | RECNAM |
| | RNGLOS | RPLCTN | RSPTIM | RUNDAT | SMOKED | SYMBOL | TARGET |
| | TIME | UVALUE | WEATHR | | | | |
| LOSCHK | FLAG | KTEST | RANDOC | RANDOM | RANGE | RPLCTN | TARGET |
| | TIME | | | | | | |
| MINCRN | HIT | KTEST | RANDOM | RANGE | RPLCTN | TARGET | TIME |
| NOREC | ICODE | | | | | | |
| NUMRIC | SYMBOL | | | | | | |
| OUTPUT | ABORT | ABRLBL | DODF | FLAG | KTEST | RPLCTN | RUNDAT |
| | SCCSS | | | | | | |
| PECHK | FLAG | HIT | KTEST | RANDOM | RANGE | RPLCTN | RSPTIM |
| | TARGET | TIME | WEATHR | | | | |
| PENAME | XVALUE | | | | | | |
| PKCHK | HIT | KTEST | RANDOM | RPLCTN | SCCSS | TARGET | |
| PPLOT | RUNDAT | | | | | | |
| PSMOKE | SMOKED | WEATHR | | | | | |
| REINTZ | ABORT | KTEST | RANDOM | RPLCTN | SCCSS | | |
| RNDREL | KTEST | MISC | RANDOM | RPLCTN | | | |
| RNGCHK | KTEST | RANGE | RPLCTN | | | | |
| SEPREC | COMENT | LOGFLG | SYMBOL | | | | |
| SMOKE | KTEST | RANDOM | RPLCTN | SMOKED | WEATHR | | |
| TIMCHK | FLAG | KTEST | RPLCTN | TIME | | | |
| TITLE | RUNDAT | STITLE | | | | | |
| USET | DVALUE | LOGFLG | OVER | UVALUE | | | |
| VISCHK | KTEST | RANGE | RPLCTN | | | | |
| WARNDC | FLAG | KTEST | RANDOM | RPLCTN | RSPTIM | TIME | |

TABLE 10-2 SUBPROGRAMS OF COPE IN WHICH EACH COMMON BLOCK OCCURS

| COMMON | SUBPROGRAMS |
|---|---|
| ABORT | ABRTTL INITLZ INPUT OUTPUT REINTZ |
| ABRLBL | BDATA1 OUTPUT |
| ACHAR | BDATA1 FINDIT INITLZ |
| BAIL | BLOCHK COPE ECHO HEADER INPUT |
| COMENT | COMMNT HEADER INPUT SEPREC |
| DESRNG | BDATA1 INPUT |
| DISPLY | BDATA1 HEADER INPUT |
| DODF | COPE CREAD DFCHK ECHO INPUT OUTPUT |
| DVALUE | DFAULT INITLZ INPUT USET |
| FLAG | ABRTTL COPE DETCTN ECHO GETTIM HEADER INITLZ INPUT LOSCHK OUTPUT PECHK TIMCHK WARNDO |
| FLTTIM | BDATA1 INPUT |
| FSRESP | BDATA1 GETTIM |
| HEADNG | BDATA1 HEADER |
| HIT | CREAD ECHO HITCHK INPUT MINTRN PECHK PKCHK |
| ICODE | FINDIT INITLZ INPUT NOREC |
| INDEX1 | FINDIT INITLZ INPUT |
| KTEST | ARTCHK BLOCHK DETCTN DFCHK DUST GETTIM HITCHK LOSCHK MINTRN OUTPUT PECHK PKCHK REINTZ RNDREL RNGCHK SMOKE TIMCHK VISCHK WARNDO |
| LOGFLG | BDATA1 ECHO GETTIM HEADER INPUT SEPREC USET |
| MISC | ARTCHK DUST ECHO HEADER INPUT RNDREL |
| OVER | ECHO HEADER INPUT USET |
| PEDESC | ECHO HEADER INPUT |
| POINT | BDATA1 INPUT |
| RANDOC | CREAD ECHO INPUT LOSCHK |
| RANDOM | ARTCHK DETCTN DFCHK DUST GETRNG GETTIM GETVIS HITCHK INPUT LOSCHK MINTRN PECHK PKCHK REINTZ RNDREL SMOKE WARNDO |
| RANGE | BLOCHK DFCHK ECHO GETRNG GETVIS HEADER HITCHK INPUT LOSCHK MINTRN PECHK RNGCHK VISCHK |

TABLE 10-2 SUBPROGRAMS OF COPE IN WHICH EACH COMMON BLOCK OCCURS - CONT'D

| COMMON | SUBPROGRAMS | | | | | | |
|---|---|---|---|---|---|---|---|
| RECNAM | ECHO | HEADER | INPUT | | | | |
| RNGLOS | CREAD | ECHO | GETRNG | INPUT | | | |
| RPLCTN | ABRTTL | ARTCHK | BLOCHK | COPE | DETCTN | DFCHK | DUST |
| | ECHO | GETRNG | GETTIM | GETVIS | HEADER | HITCHK | INPUT |
| | LOSCHK | MINTRN | OUTPUT | PECHK | PKCHK | REINTZ | RNUREL |
| | RNGCHK | SMOKE | TIMCHK | VISCHK | WARNDO | | |
| RSPTIM | CREAD | DETCTN | ECHO | GETTIM | HEADER | HITCHK | INPUT |
| | PECHK | WARNDO | | | | | |
| RUNDAT | CCPLOT | COPE | DFAULT | ECHO | FINDIT | HEADER | INITLZ |
| | INPUT | OUTPUT | PPLOT | TITLE | | | |
| SCCSS | OUTPUT | PKCHK | REINTZ | | | | |
| SMOKED | BDATA1 | ECHO | HEADER | INPUT | PSMOKE | SMOKE | |
| STITLE | BDATA1 | TITLE | | | | | |
| SYMBOL | BDATA1 | FINDIT | HEADER | IDCHAR | INPUT | NUMRIC | SEPREC |
| TARGET | BLOCHK | DFCHK | ECHO | GETRNG | HEADER | HITCHK | INPUT |
| | LOSCHK | MINTRN | PECHK | PKCHK | | | |
| TIME | ADDTBR | ADDTOF | BLOCHK | COPE | DETCTN | DFCHK | ECHO |
| | GETRNG | GETTIM | HEADER | HITCHK | INPUT | LOSCHK | MINTRN |
| | PECHK | TIMCHK | WARNDO | | | | |
| UVALUE | HEADER | INITLZ | INPUT | USET | | | |
| WEATHR | CREAD | DFCHK | ECHO | GETRNG | GETVIS | INPUT | PECHK |
| | PSMOKE | SMOKE | | | | | |
| XVALUE | BDATA1 | PENAME | | | | | |

TABLE 10-3 COPE EQUIVALENCE CLASSES

| NAME (Name of First Item) | LENGTH | MEMBERS Bias | Name | Length |
|---|---|---|---|---|
| ACQDAT | 161 | 0 | NRP | 1 |
| | | 1 | NPP | 1 |
| | | 2 | NRNGCL | 1 |
| | | 3 | CRNGD | 22 |
| | | 25 | SEGLOC | 121 |
| | | 146 | RNGCLB | 11 |
| | | 157 | VELTBL | 4 |
| (Occurs in subroutines CREAD and INPUT) | | | | |
| AINVDA | 167 | 0 | DLTT | 6 |
| | | 6 | PKTBL | 20 |
| | | 26 | NRNGTT | 1 |
| | | 27 | RNGTTF | 20 |
| | | 47 | TTF | 120 |
| (Occurs in subroutines CREAD and INPUT) | | | | |
| D | 70 | 0 | DVALUE | 70 |
| (Occurs in subroutine INPUT) | | | | |
| DFSDAT | 31 | 0 | DFDOKL | 30 |
| | | 30 | NDFSP | 1 |
| (Occurs in subroutine CREAD and INPUT) | | | | |
| ICODE | 1 | 0 | CODE | 1 |
| (Occurs in subroutines FINDIT, INITLZ, and INPUT) | | | | |
| PEDATA | 4260 | 0 | INDEX | 60 |
| | | 60 | PETBL | 4200 |
| (Occurs in subroutines CREAD and INPUT) | | | | |

TABLE 10-3 COPE EQUIVALENCE CLASSES - CONT'D

| NAME (Name of First Item) | LENGTH | MEMBERS Bias | Name | Length |
|---|---|---|---|---|
| PSTDAT | 41 | 0 | NRNGPS | 1 |
| | | 1 | RNGPST | 10 |
| | | 11 | PSTTBL | 30 |
| (Occurs in subroutines CREAD and INPUT) | | | | |
| RECODE | 8 | 0 | WCODE | 1 |
| | | 1 | ACCODE | 1 |
| | | 2 | RSCODE | 1 |
| | | 3 | DFCODE | 1 |
| | | 4 | ROCODE | 1 |
| | | 5 | PECODE | 1 |
| | | 6 | PSCODE | 1 |
| | | 7 | AICODE | 1 |
| (Occurs in subroutine (HEADER) | | | | |
| RODATA | 22 | 0 | TCRIT | 1 |
| | | 1 | NPLOSR | 1 |
| | | 2 | RNGPLS | 10 |
| | | 12 | PRBLOS | 10 |
| (Occurs in subroutines CREAD and INPUT) | | | | |
| RSPDAT | 49 | 0 | BATRTM | 3 |
| | | 3 | XMVADF | 2 |
| | | 5 | DUM1 | 1 |
| | | 6 | BCSPTM | 4 |
| | | 10 | XMTTIM | 4 |
| | | 14 | TMEAN | 4 |
| | | 18 | TSIGMA | 4 |
| | | 22 | DETTMA | 20 |
| | | 42 | NDT | 1 |
| | | 43 | TRARRY | 6 |
| (Occurs in subroutines CREAD and INPUT) | | | | |

TABLE 10-3 COPE EQUIVALENCE CLASSES - CONT'D

| NAME (Name of First Item) | LENGTH | Bias | MEMBERS Name | Length |
|---|---|---|---|---|
| U | 70 | 0 | UVALUE | 70 |
| (Occurs in subroutines HEADER and INPUT) | | | | |
| WDATA | 193 | 0 | PRCLCG | 1 |
| | | 1 | PRCFLS | 2 |
| | | 3 | PRGCFL | 22 |
| | | 25 | W | 132 |
| | | 157 | VIS | 11 |
| | | 168 | CLOUD | 6 |
| | | 174 | NCC | 1 |
| | | 175 | NVL | 1 |
| | | 176 | PASQL | 6 |
| | | 182 | WNDSPD | 9 |
| | | 191 | HUMID | 2 |
| (Occurs in subroutines CREAD and INPUT) | | | | |



Next page is blank.

CHAPTER 11

# 11. FLOWCHARTS

This chapter includes flowcharts of the COPE main program and subroutine INPUT. Flowcharts of the other subroutines are not included since they are generally simple enough to understand by reading the descriptions in chapter 9 and studying the actual program code with the aid of the glossary of variables in chapter 12.

Several conventions are observed in the flowcharts. First, connectors on-page are indicated by circles whereas connectors off-page are indicated by pentagons. Numbered connectors correspond to actual statement numbers in the program code whereas letter connectors are merely flowcharting conveniences. Each STOP block in the flowchart includes the section number of this report that explains that stop.

A list of abbreviations used in the flow charts is included on the next page.

ABBREVIATIONS USED IN FLOW CHARTS

| | |
|---|---|
| acq | Acquisition |
| CLGP | COPPERHEAD |
| C.O. | Completely obscured |
| comt | comment |
| desig | designator |
| DF | direct fire |
| dgt | digital |
| D.O. | designator operator |
| F.E. | fully exposed |
| fn | function |
| H.D. | hull defilade |
| hdng | heading |
| KW | keyword |
| max | maximum |
| mo | month |
| msg | message |
| msn | mission |
| no | number |
| nt | night |
| opt | option |
| OW | Option word |
| param | parameterized |
| pr | priority |
| prob | probability |
| rd(s) | round(s) |
| resp | response |
| R.N. | random number(s) |
| rng | range |
| SG | shooting gallery |
| smk | smoke |
| supp | suppression |
| temp | temporary |
| tgt | target |
| vel | velocity |
| vis | visibility |
| w/ | with |
| # | number |

Flowchart of Main Program of COPE

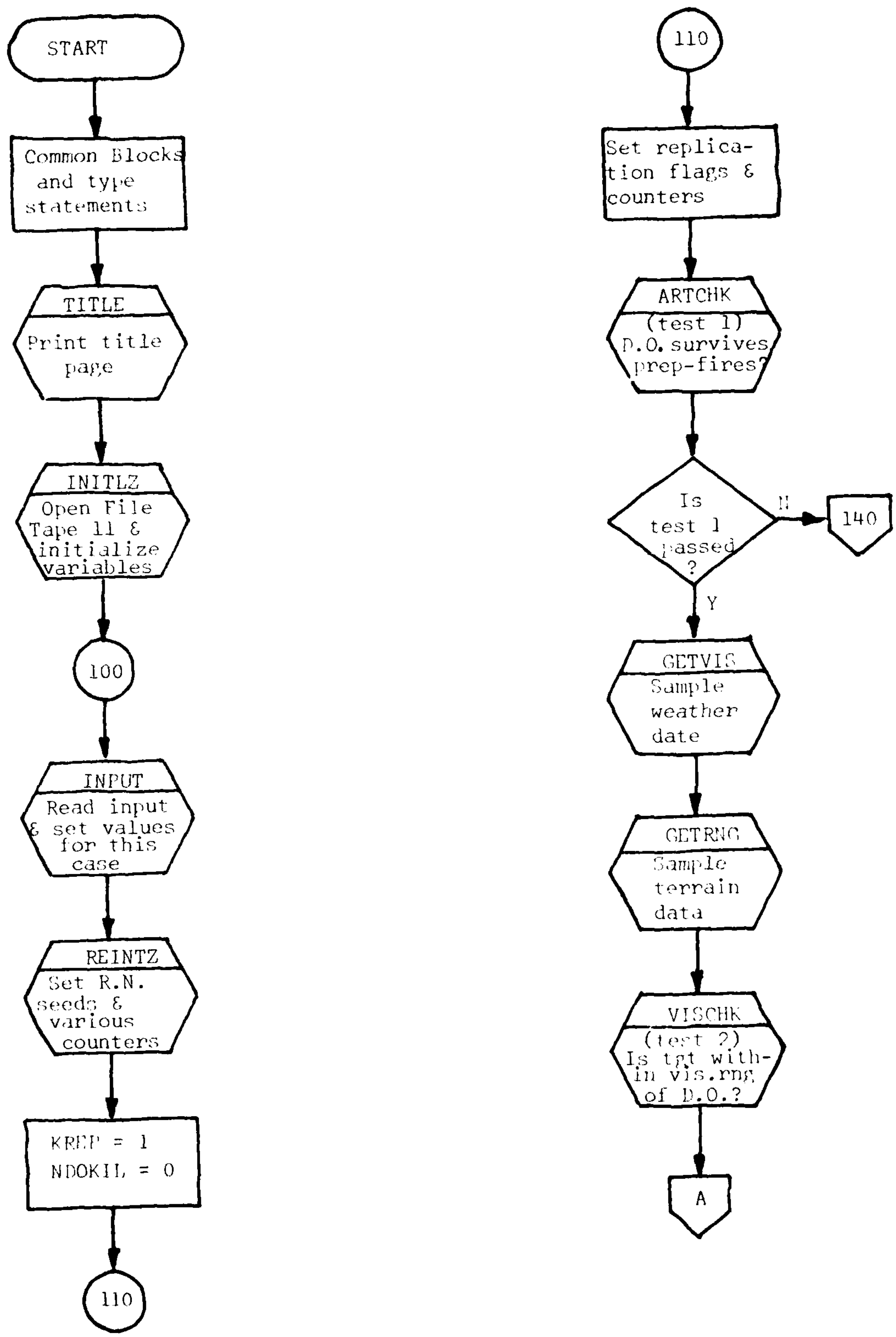



FIGURE 11-1 Flowchart of Main Program of COPE (Page 1 of 5)

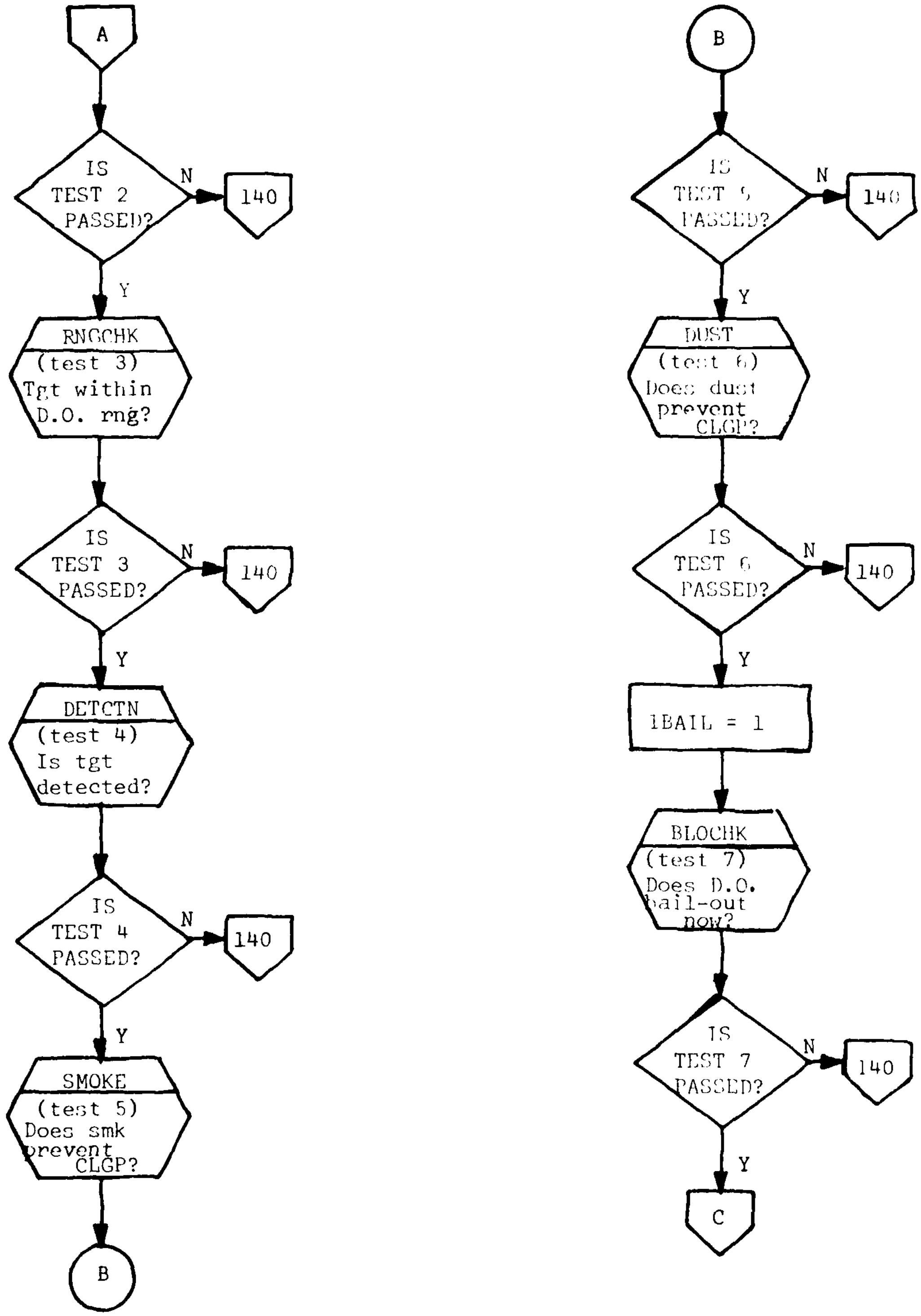



FIGURE 11-1 Flowchart of Main Program of COPE (Page 2 of 5)

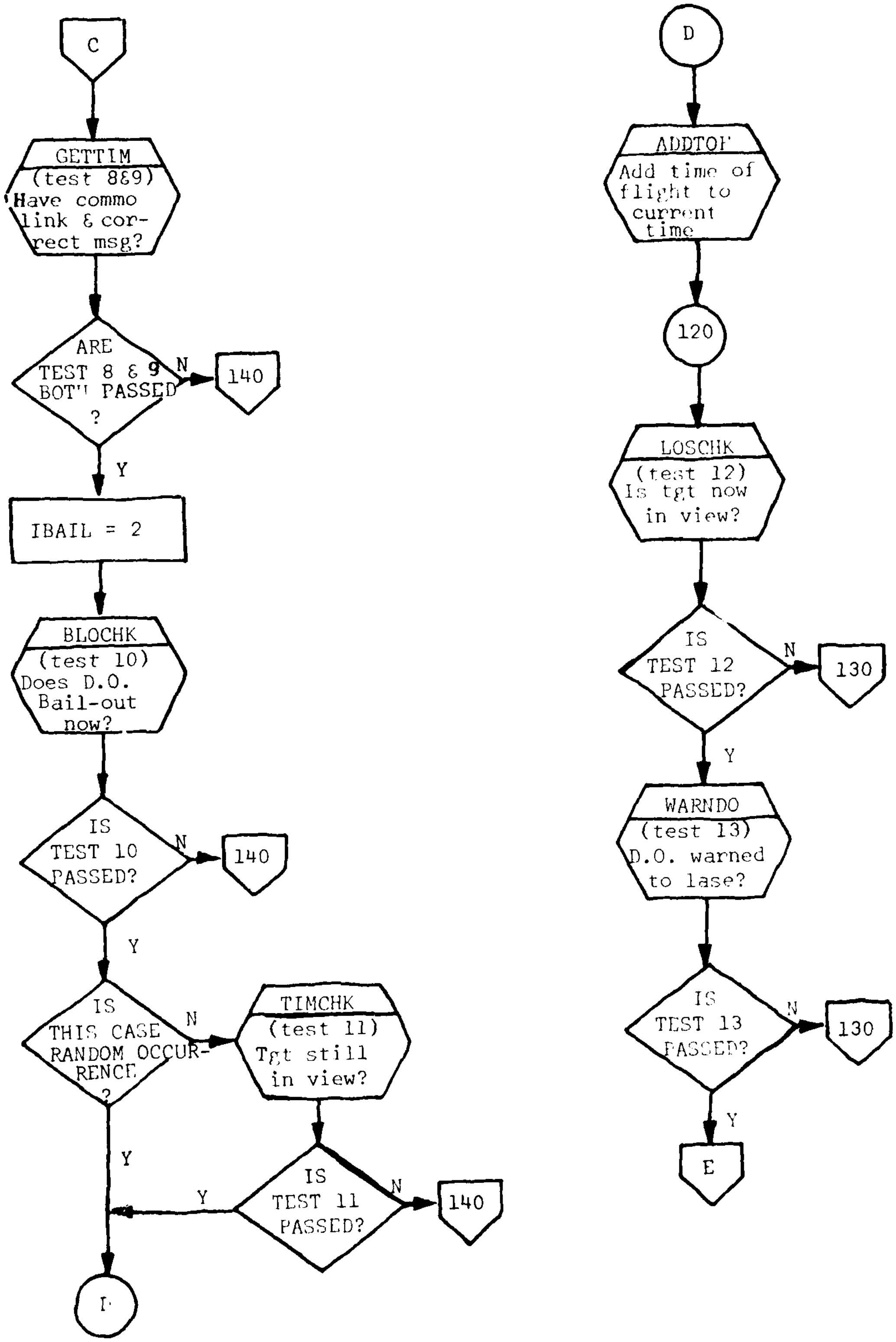



FIGURE 11-1 Flowchart of Main Program of COPE (Page 3 of 5)

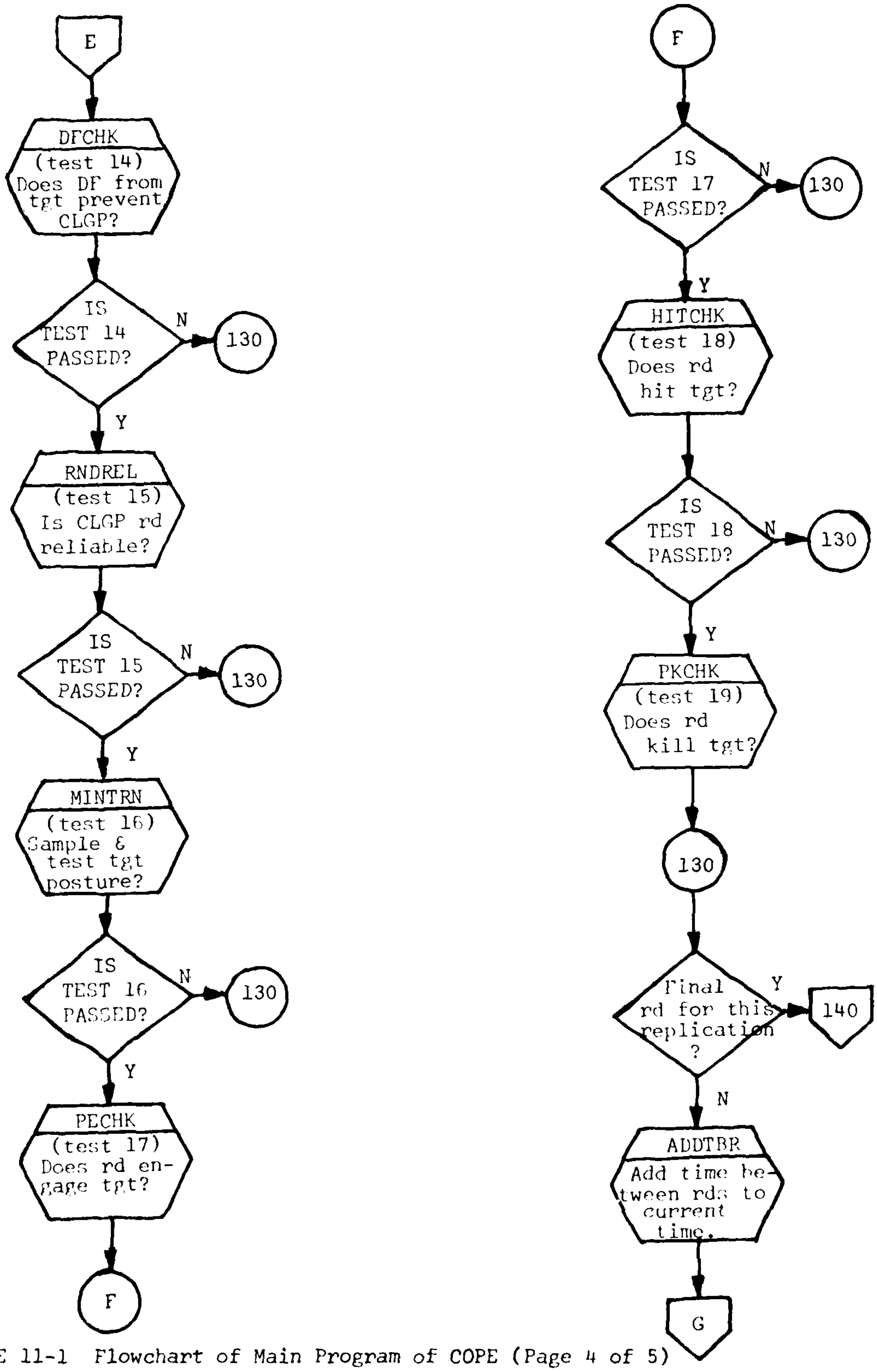



FIGURE 11-1 Flowchart of Main Program of COPE (Page 4 of 5)

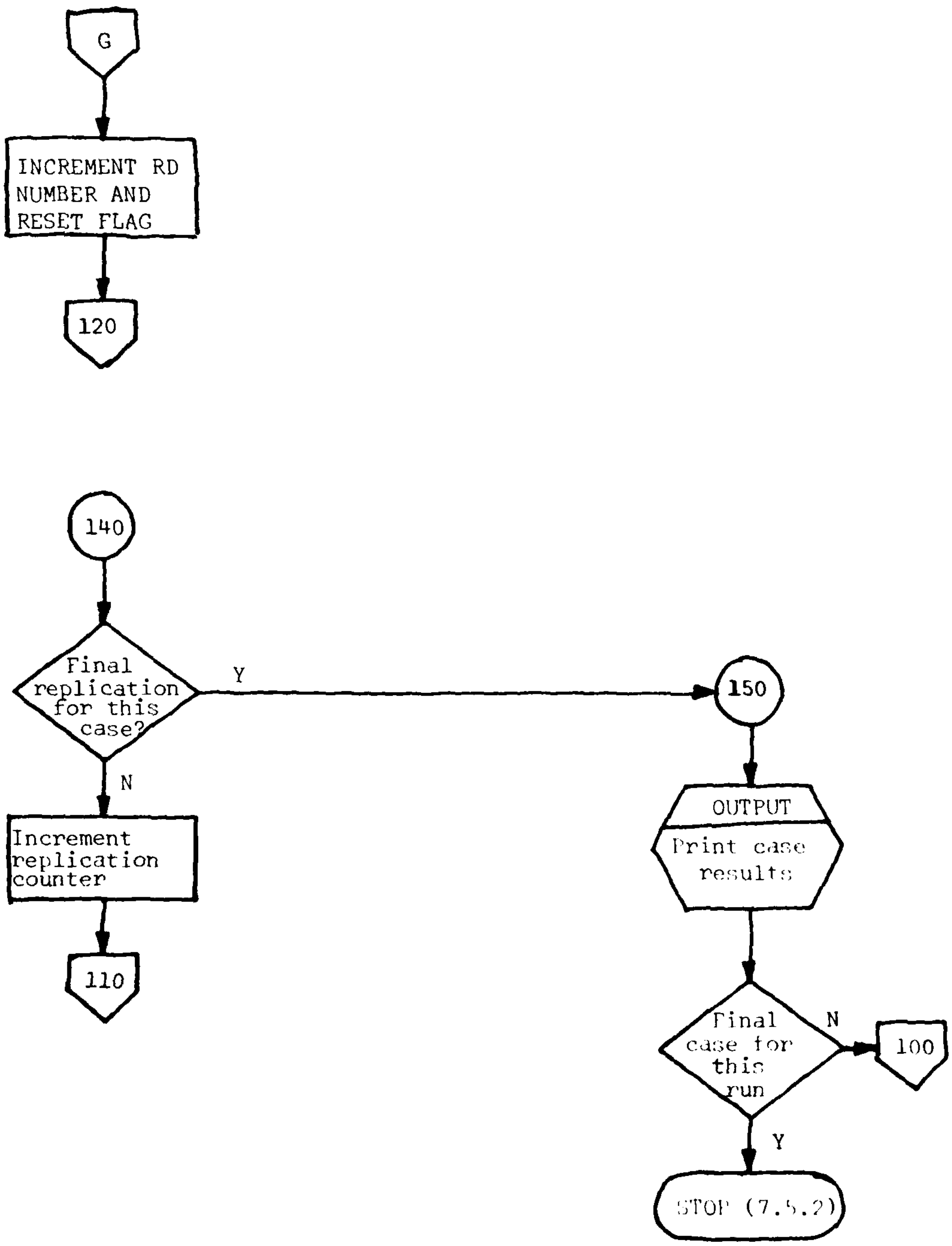



FIGURE 11-1 Flowchart of Main Program of COPE (Page 5 of 5)

Flowchart of Subroutine INPUT of COPE

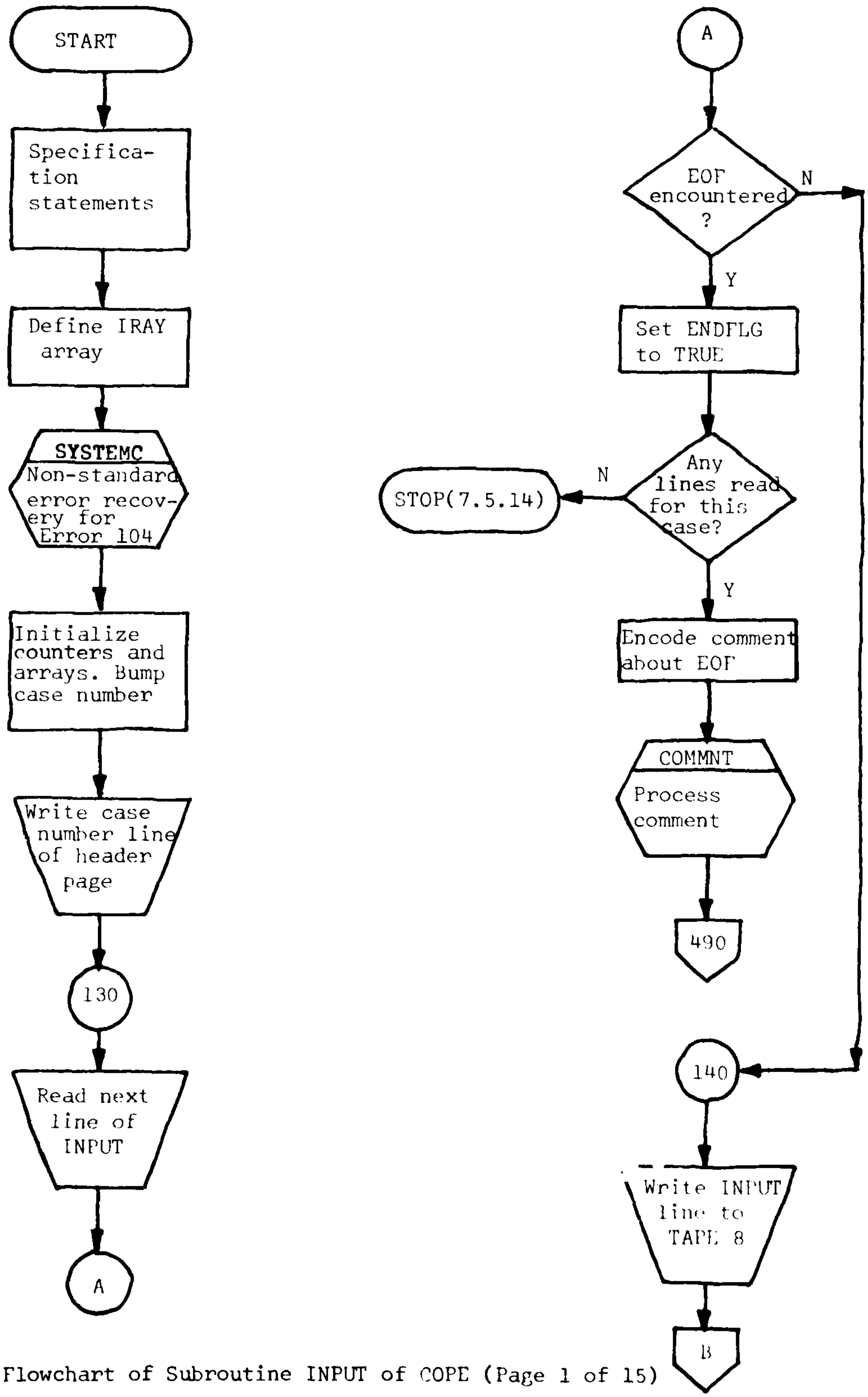



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 1 of 15)

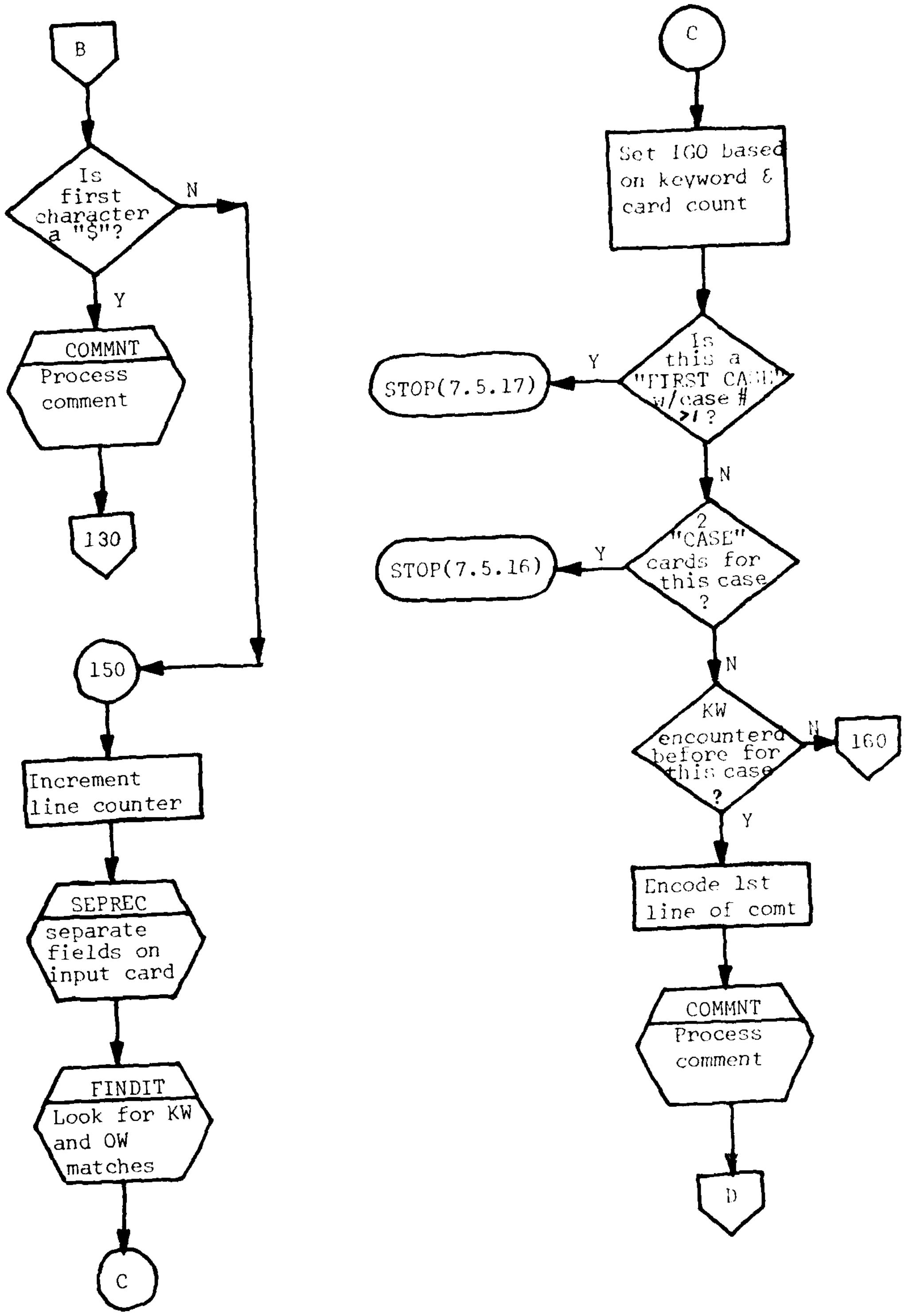



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 2 of 15)

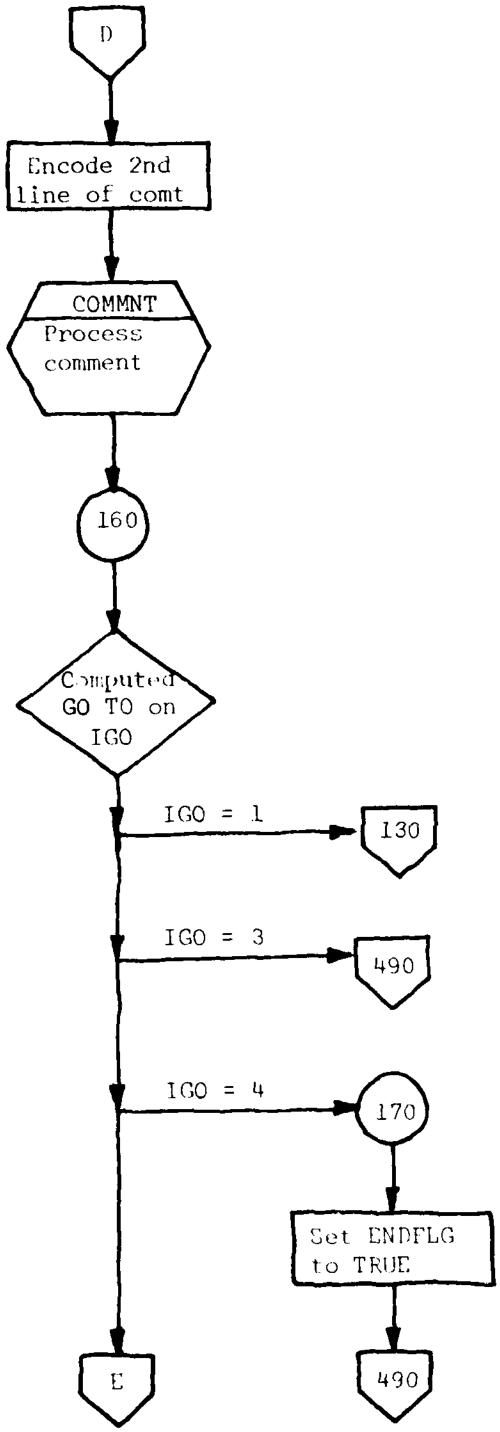



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 3 of 15)

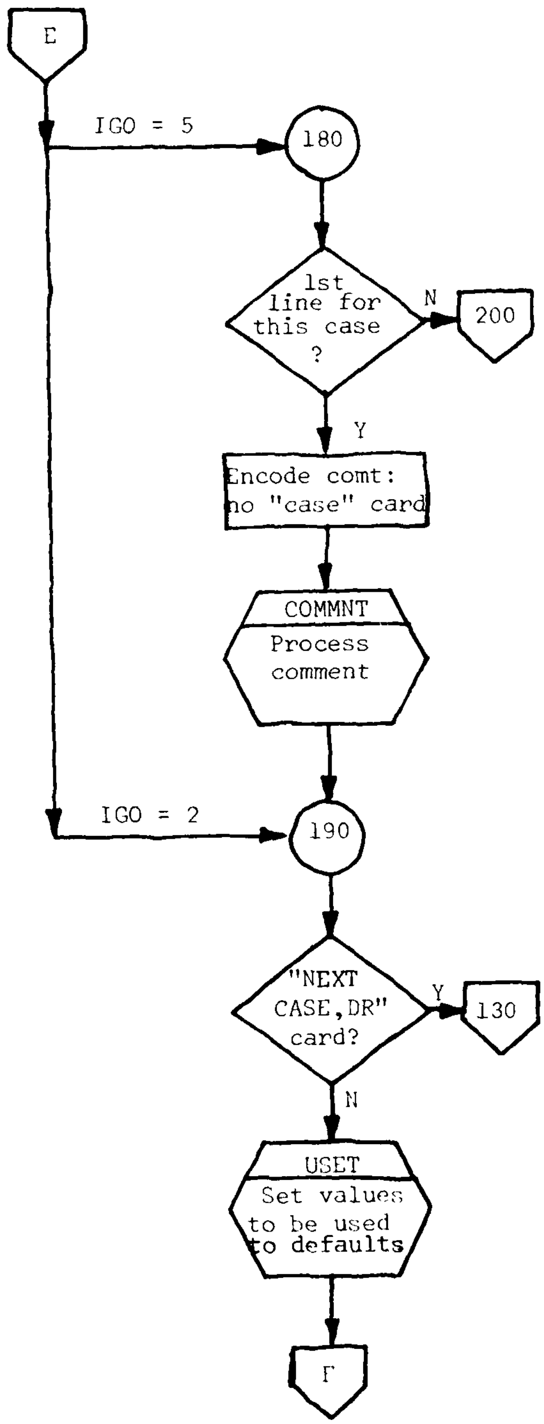



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 4 of 15)

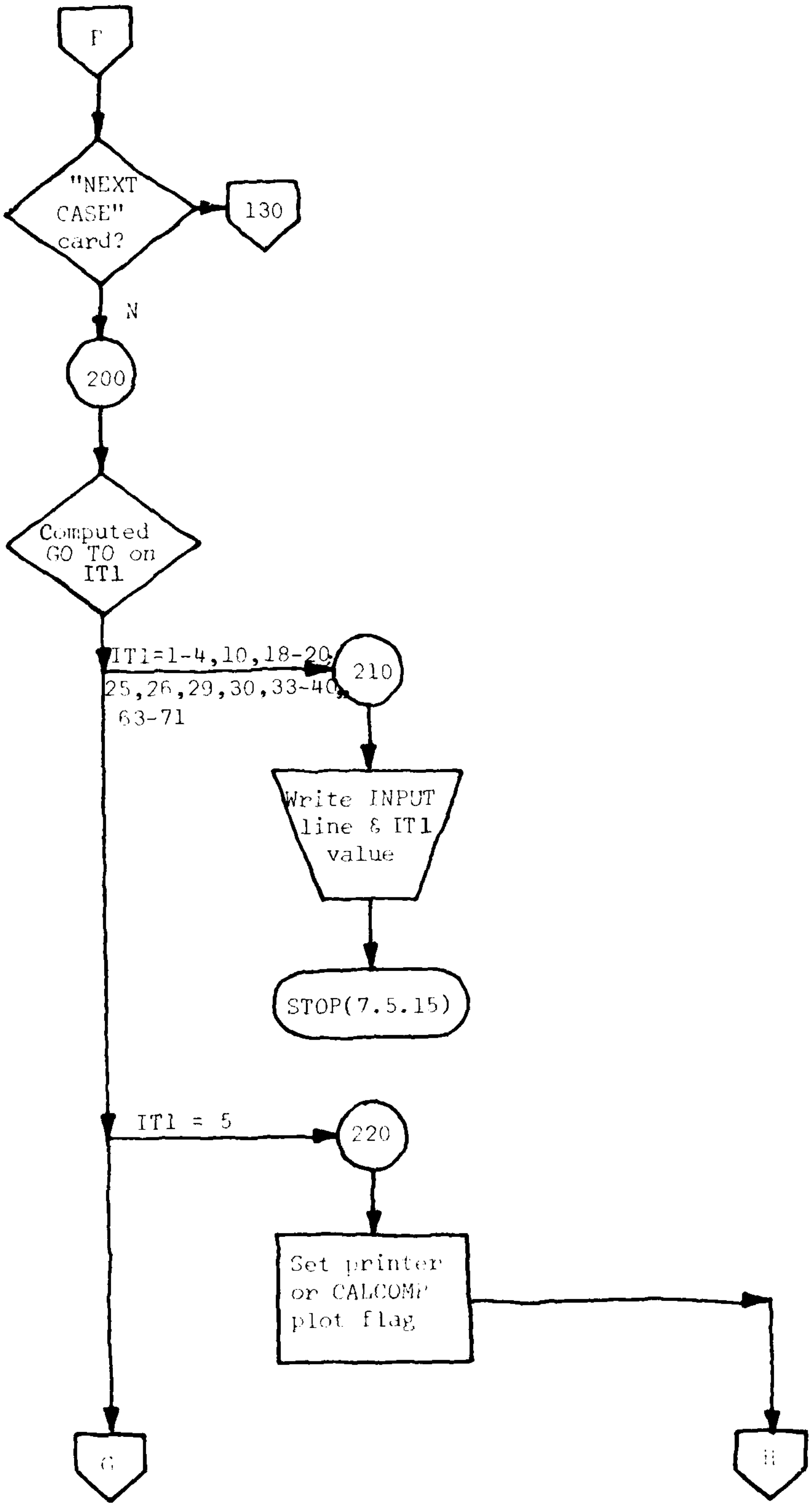



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 5 of 15)

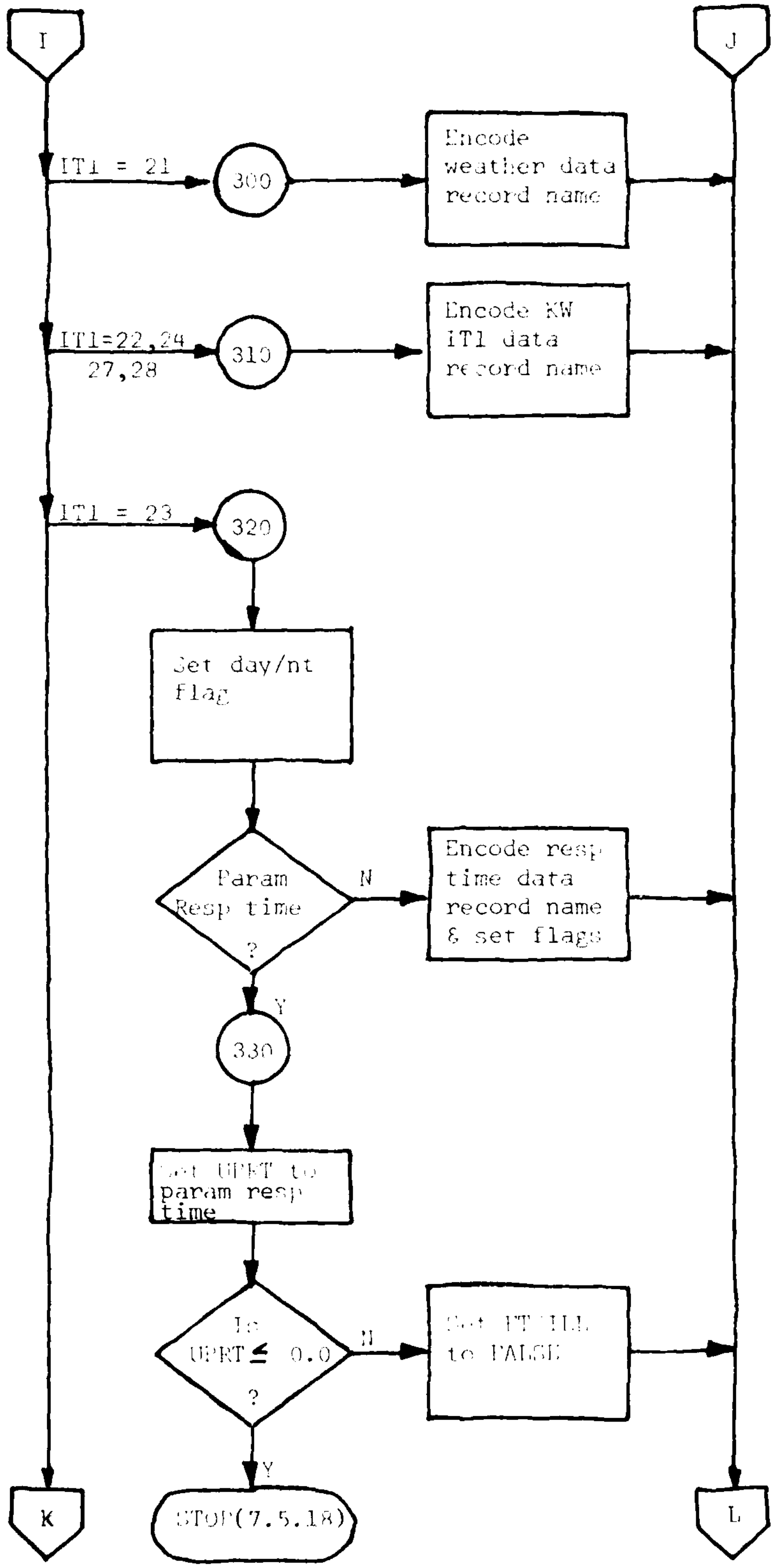



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 7 of 15)

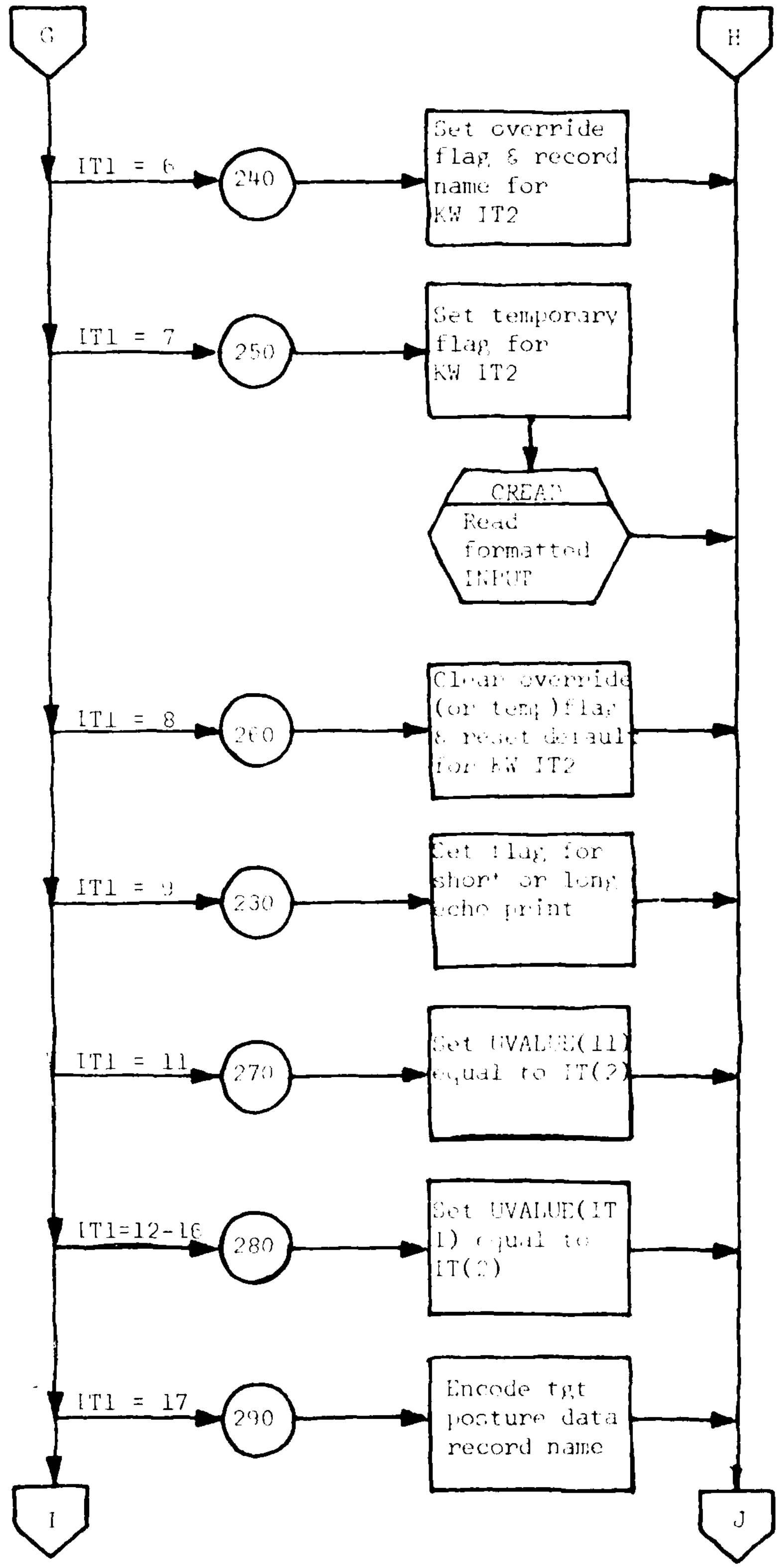



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 6 of 15)

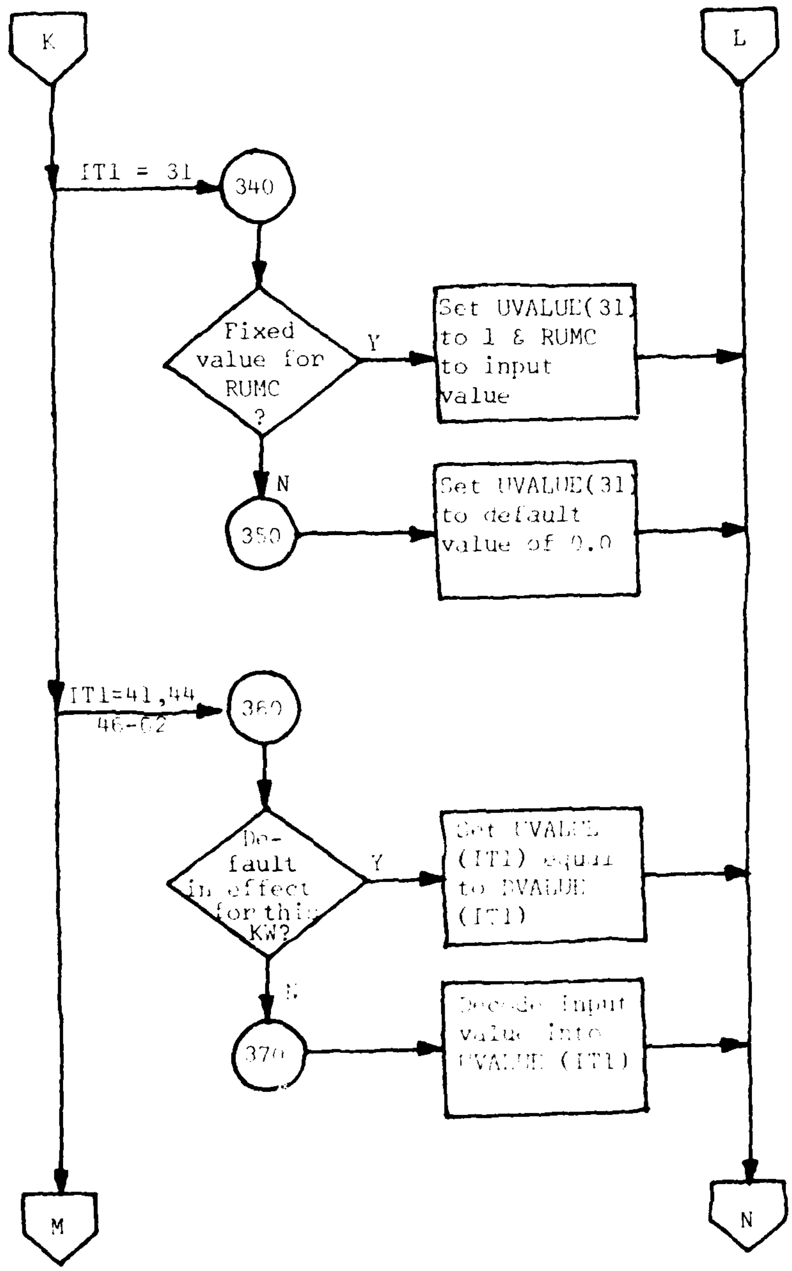



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 8 of 15)

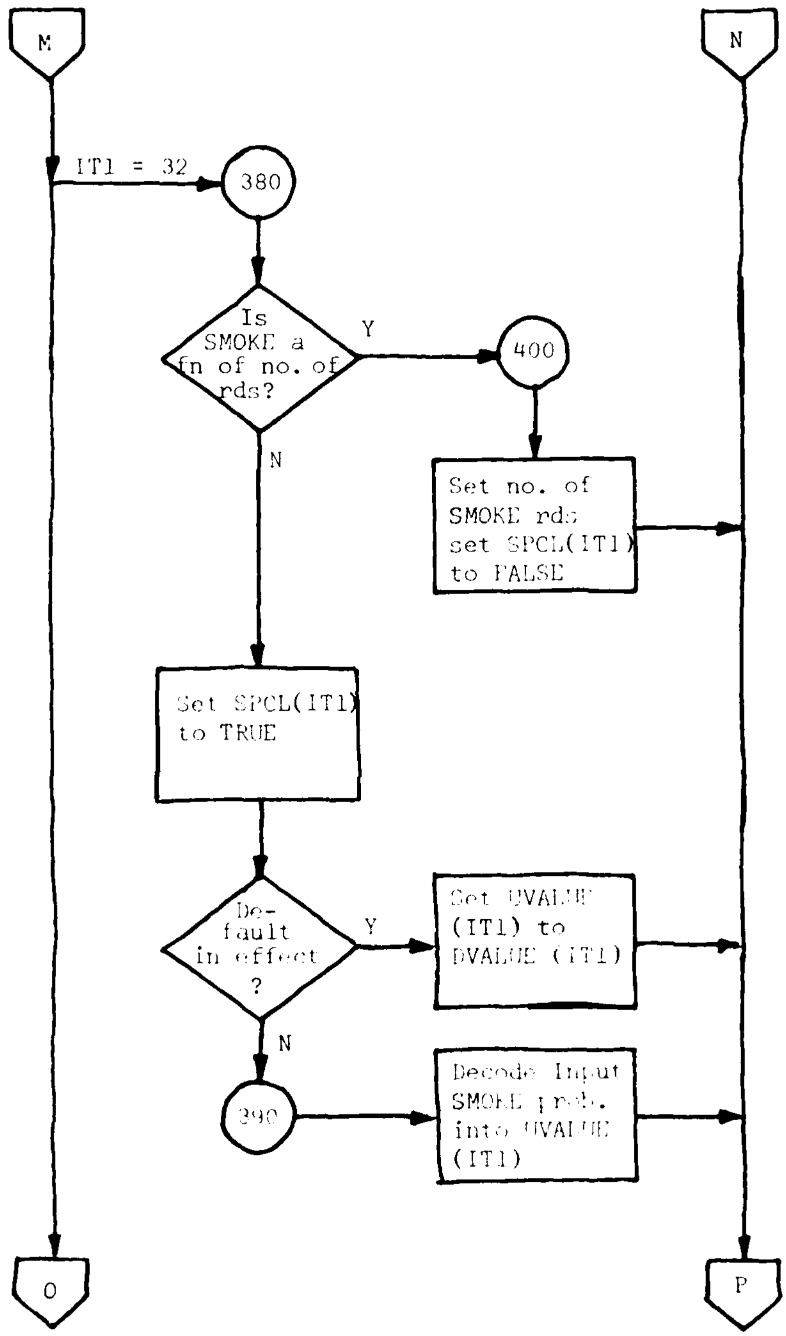



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 9 of 15)

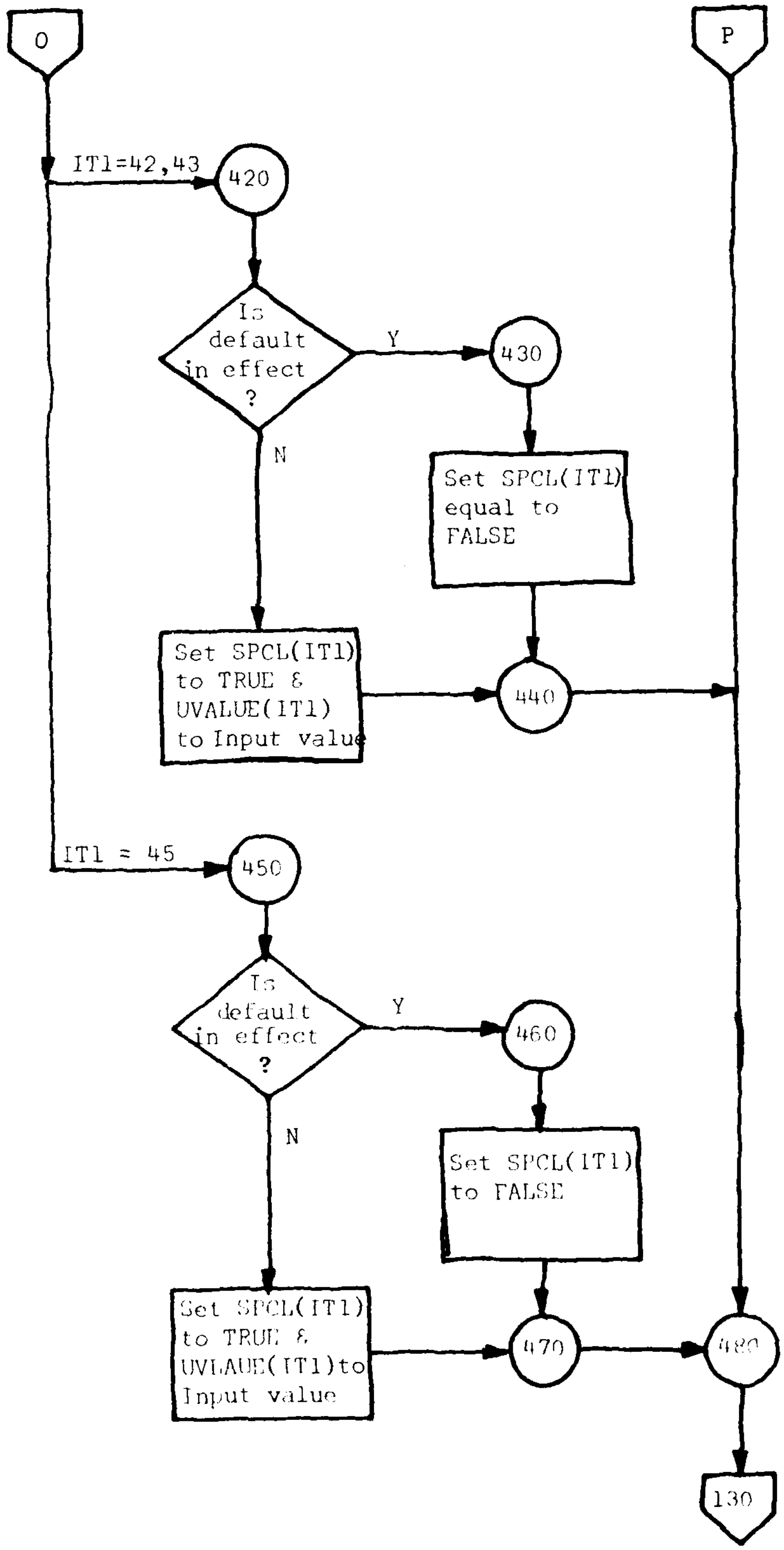



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 10 of 15)

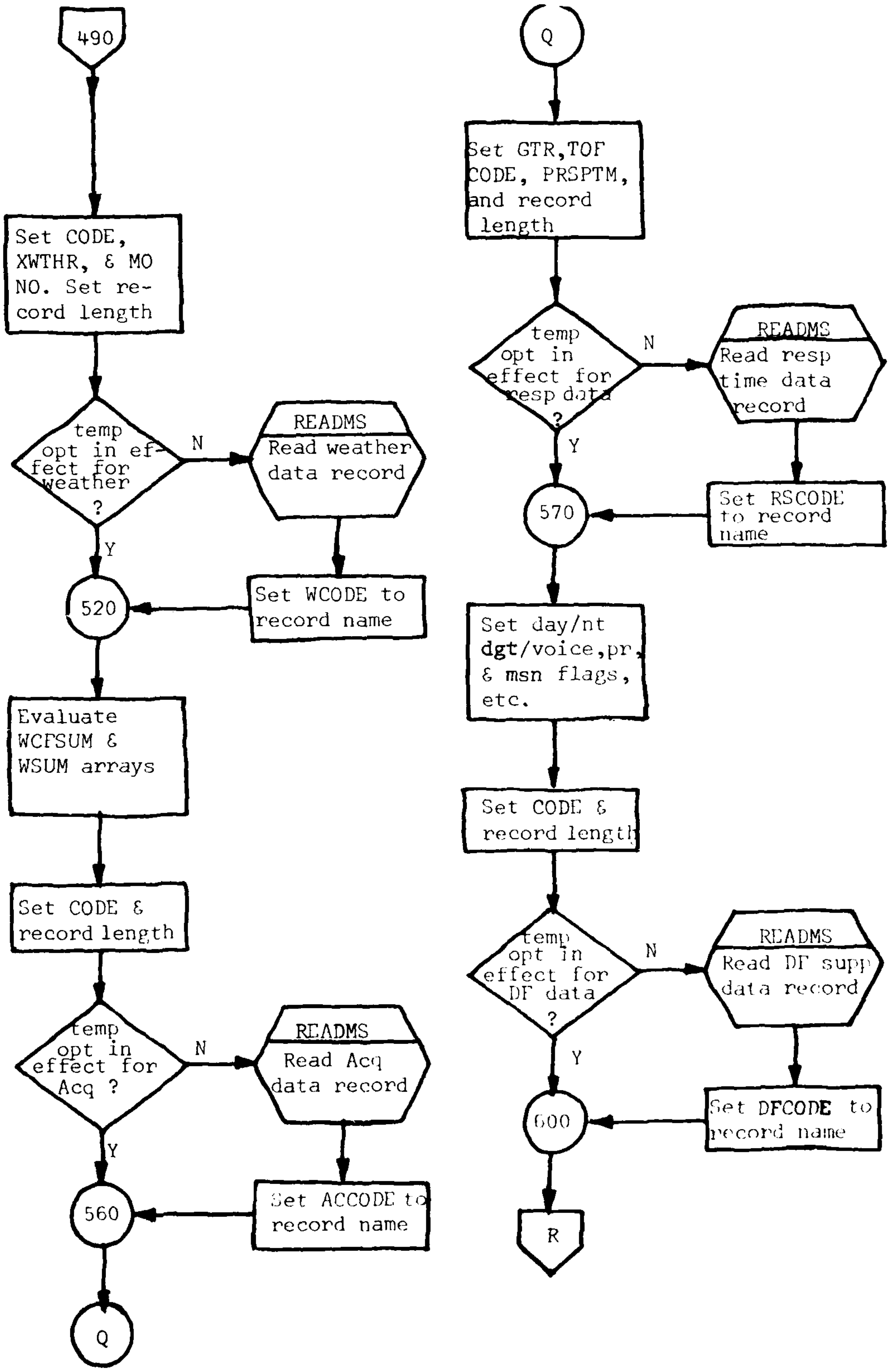



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 11 of 15)

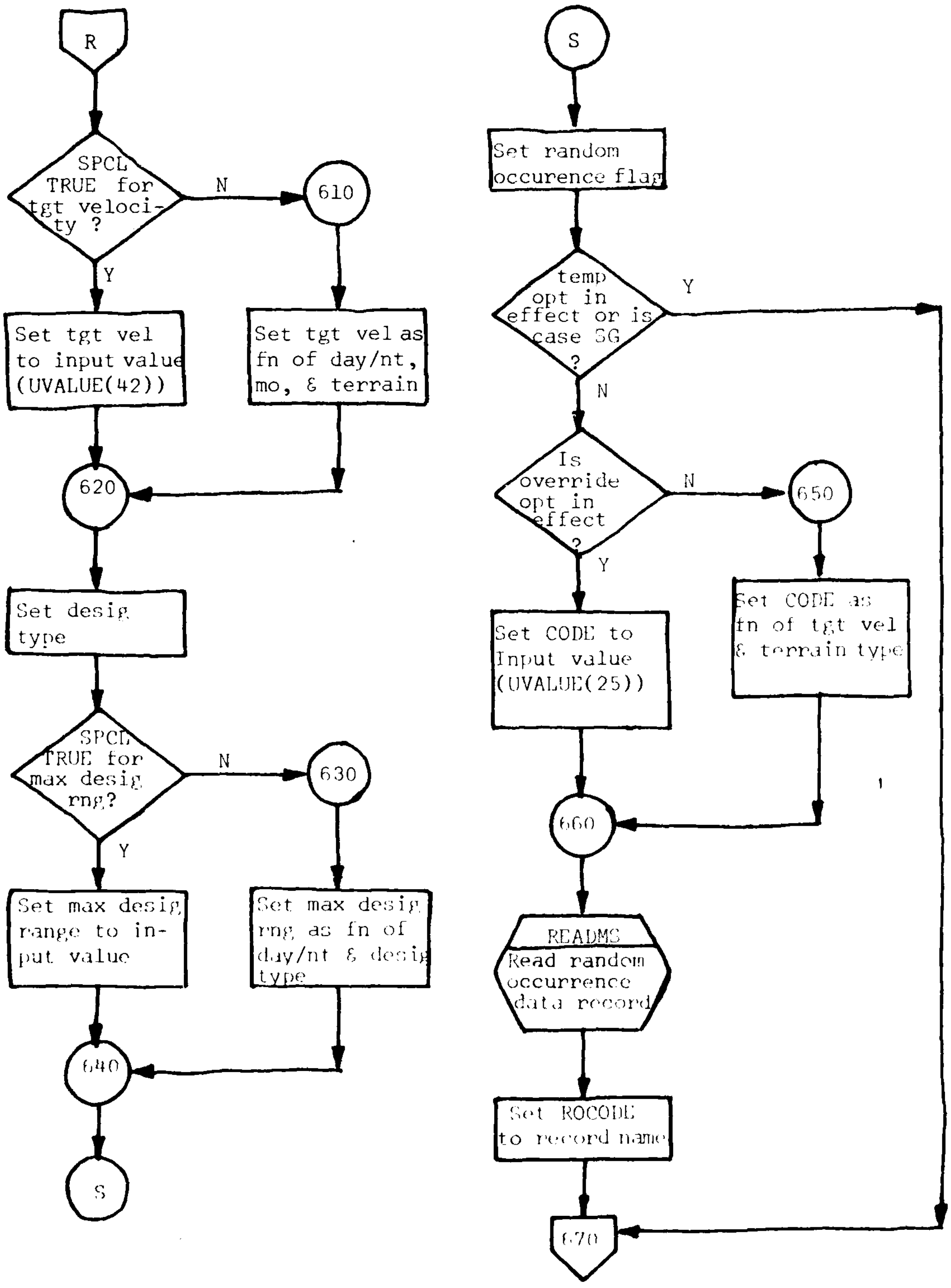



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 12 of 15)

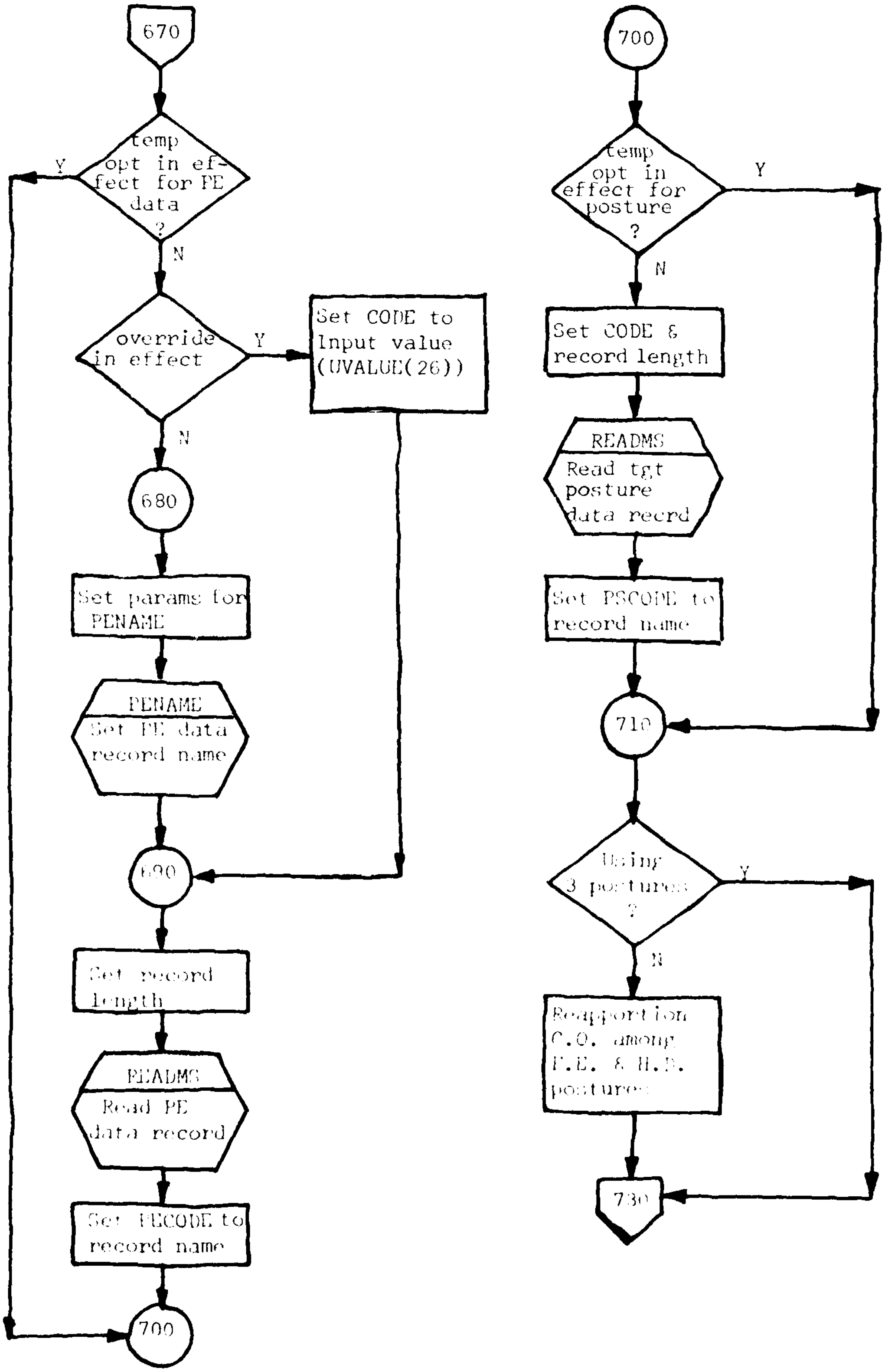



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 13 of 15)

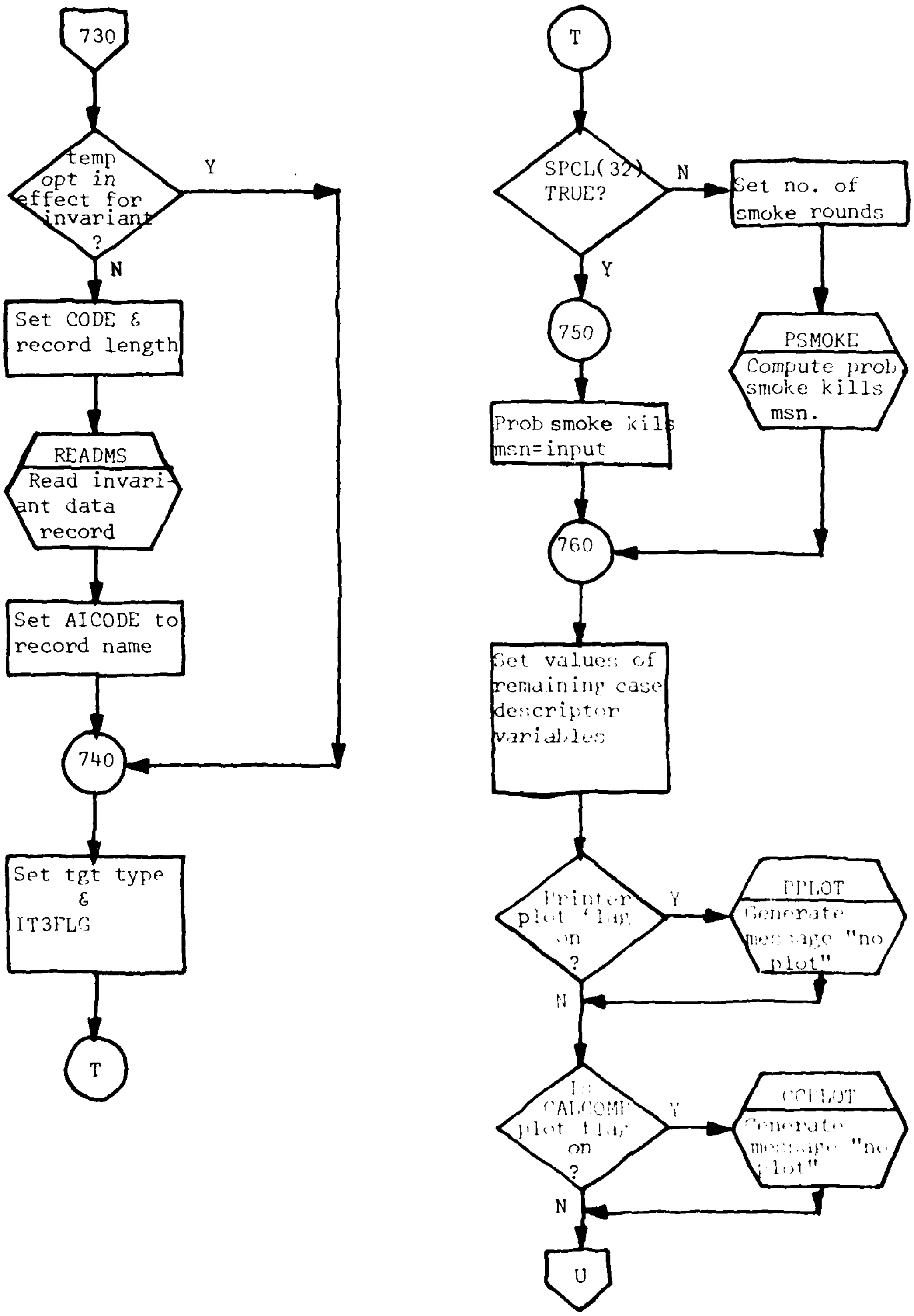



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 14 of 15)

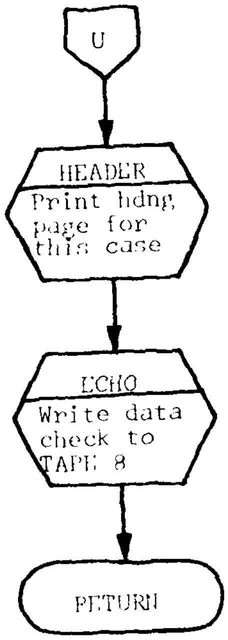



FIGURE 11-2 Flowchart of Subroutine INPUT of COPE (Page 15 of 15)

# CHAPTER 12

## 12. GLOSSARY OF MAIN COPE PROGRAM VARIABLES

This chapter consists of a glossary of variables for the main COPE program. (Glossaries for the PREPMS and PRBLOS preprocessor programs are respectively, included in sections 14.3 and 15.3. No glossary is provided in this report for the PAM preprocessor program.)

The glossary consists of an alphabetical list of the variable names. For each variable name, there is an entry giving VARIABLE, TYPE, COMMON BLOCK, UNITS, and DEFINITION.

Under the VARIABLE heading is the variable name and, if the variable is an array, the array dimensions are given in parentheses after the name.

Under the TYPE column, the variable type is given; if the type is omitted, it means the FORTRAN type default is in effect (i.e., a variable beginning with letter I, J, K, L, M, or N is integer type whereas one beginning with any other letter is single precision real type).

Under the COMMON BLOCK column is the name of the common block (if any) to which the variable belongs. If this column contains "(LOCAL)", it means the variable is a local variable; if it contains "(F.P.)", it means the variable is a formal parameter in a FUNCTION or SUBROUTINE.

Under the UNITS column is the name of the units of measure in which the variable is to be given. If a blank occurs in this column, it means that the variable is unitless.

Finally, under the DEFINITION heading is a brief description of the variable.

GLOSSARY OF MAIN CODE PROGRAM VARIABLES

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| A(80)<br>A(N) | | (LOCAL)<br>(F.P.) | | ARRAY INTO WHICH INPUT LINE IS READ BY SUBROUTINE INPUT. A(I) IS THE ITH CHARACTER IN THE INPUT LINE. ALSO USED AS A FORMAL PARAMETER TO RECEIVE COMMENTS IN SUBROUTINE COMMNT OR TO PASS INPUT LINE TO SUBROUTINE SEPREC TO UNSTRING KEYWORDS AND OPTION WORDS. |
| ABRPCT(20, 6) | | (LOCAL) | | ABRPCT(I, J) IS THE PERCENT OF THE TOTAL NUMBER OF REPLICATIONS THAT ABORTED FOR ITH ABORT REASON ON ROUND J. IF ABRPCT(I, J) IS UNDEFINED, IT IS SET EQUAL TO -1. |
| ACCODE | | RECNAM | | RECORD NAME KEY FOR THE ACQUISITION DATA RECORD LAST READ FROM THE WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| ACQDAT(161) | | (LOCAL) | | ARRAY TO WHICH ELEMENTS OF ACQUISITION RANGE DATA BLOCK ARE EQUIVALENCED. |
| ACQLBL(2, 4) | | HEADNG | | ALPHANUMERIC DESCRIPTOR OF THE TERRAIN TYPE USED FOR CURRENT CASE. IT IS USED TO LABEL THE "TERRAIN AND LOS" BLOCK OF THE OUTPUT HEADING. |
| AICODE | | RECNAM | | RECORD NAME KEY FOR THE "INVARIANT" DATA RECORD LAST READ FROM THE WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| AINVDA(167) | | (LOCAL) | | ARRAY TO WHICH ELEMENTS OF INVARIANT DATA BLOCK ARE EQUIVALENCED. |
| ALF | | (LOCAL) | | TEMPORARY NAME FOR THE ALPHA PARAMETER OF THE GAMMA DISTRIBUTION USED TO SAMPLE DIGITAL MESSAGE DEVICE DELAY TIME. |
| ALFBET(27) | | SYMBOL | | ALFBET(I) IS THE ITH LETTER OF THE ALPHABET FOR I=1,2,...,26 ALFBET(27) IS AN APOSTROPHE. THESE SYMBOLS ARE ALL STORED IN A1 FORMAT. |
| ALPHA | | (F.P.) | | ALPHA IS THE 'SHAPING FACTOR' OF THE GAMMA DISTRIBUTION. |
| ALPHA(2, 2) | | RSPTIM | | ALPHA(I, J) IS THE ALPHA PARAMETER TO BE USED WITH THE GAMMA DISTRIBUTION (I=1 FOR DAY, I=2 FOR NIGHT, J=1 FOR A PRE-PLANNED TARGET, AND J=2 FOR A TARGET OF OPPORTUNITY). |
| ANGLET | | PLDESC<br>(F.P.) | DEGREES | THE ANGLE BETWEEN THE GUN-TARGET LINE AND THE DESIGNATOR-TARGET LINE. |
| ANUMBR(11) | | SYMBOL | | ANUMBR(I)=I-1 FOR $1 \le I \le 10$. ANUMBR(11) IS A MINUS SIGN. THESE ARE STORED IN A1 FORMAT. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| AOUT(4, 40) | | (LOCAL) | | ARRAY INTO WHICH THE FIRST COLUMN OF CASE HEADING OUTPUT BLOCKS ARE ENCODED PRIOR TO BEING PRINTED OUT IN SUBROUTINE HEADER. |
| AVALUE(9) | | (LOCAL) | | ARRAY USED TO TEMPORARILY STORE VALUES USED IN CREATING AND DECIPHERING PROBABILITY OF ENGAGEMENT RECORD NAME KEYS USED TO READ TAPE 11. |
| A1 | | (LOCAL) | | USED IN FUNCTION URAN31 TO COMPUTE PSEUDO-RANDOM NUMBER. |
| B(8, 10)<br>B(M, N) | | (LOCAL)<br>(F.P.) | | B(I, J) CONTAINS CHARACTERS 10*(I-1)+1 THROUGH 10*I OF THE JTH KEYWORD OR OPTION WORD OF THE MOST RECENTLY READ NON-COMMENT INPUT LINE STORED IN A10 FORMAT WITH MOST BLANKS REMOVED. |
| BATRTM(3) | | RSPTIM | SECONDS | BATRIM(I) IS THE TIME USED BY THE FIRING BATTERY FROM THE TIME IT RECEIVES A COPPERHEAD FIRE MISSION ORDER FROM THE FDC UNTIL IT IS READY TO FIRE THE FIRST COPPERHEAD ROUND OF THAT MISSION. I=1 FOR PRE-PLANNED TARGETS, I=2 FOR TARGETS OF OPPORTUNITY, I=3 FOR PRIORITY PRE-PLANNED TARGETS. |
| BCSPTM(2, 2) | | RSPTIM | SECONDS | BCSPTM(I, J) IS THE TIME REQUIRED FOR THE BATTERY COMPUTER SYSTEM TO PROCESS THE COPPERHEAD FIRE MISSION REQUEST. I=1 FOR DIGITAL COMMUNICATIONS, I=2 FOR VOICE COMMUNICATIONS; J=1 FOR PRE-PLANNED TARGETS, J=2 FOR TARGETS OF OPPORTUNITY. |
| BETA(2, 2) | | RSPTIM | SECONDS | BETA(I, J) IS THE SCALING FACTOR TO BE USED WITH THE GAMMA DISTRIBUTION (I=1 FOR DAY, I=2 FOR NIGHT; J=1 FOR PRE-PLANNED TARGETS, J=2 FOR TARGETS OF OPPORTUNITY). |
| BLANK | | SYMBOL | | BLANK IS A BLANK SYMBOL (ONE EMPTY PRINT SPACE) STORED IN A1 FORMAT. |
| BLORNG | | BAIL | METERS | BLORNG IS THE DESIGNATOR BAIL-OUT RANGE. |
| BOUT(4, 40) | | (LOCAL) | | ARRAY INTO WHICH SECOND COLUMN OF CASE HEADING OUTPUT BLOCKS ARE ENCODED PRIOR TO BEING PRINTED OUT IN SUBROUTINE HEADER. |
| C(80) | | (LOCAL) | | C(I) IS THE ITH CHARACTER OF AN OPTION WORD THAT IS BEING CHECKED TO DETERMINE WHETHER IT IS NUMERIC. (STORED IN A1 FORMAT.) |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| CHAR(10, 120) | | ACHAR | | CHAR(I, J) FOR I>2 CONTAINS CHARACTERS 10*(I-3)+1 THROUGH 10*(I-2) OF THE JTH KEYWORD CHARACTER STRING IN A10 FORMAT. CHAR(1, J) IS THE KEYWORD NUMBER AND CHAR(2, J) IS THE NUMBER OF OPTION WORD CHARACTER STRINGS ALLOWED WITH JTH KEYWORD. |
| CHARA(10, 30) | | (LOCAL) | | CHAR(I, J) FOR I>2 CONTAINS CHARACTERS 10*(I-3)+1 THROUGH 10*(I-2) OF THE JTH OPTION WORD CHARACTER STRING IN A10 FORMAT CHARA(1, J) IS THE OPTION WORD CHOICE NUMBER AND CHARA(2, J) IS THE NUMBER INDICATING THE LEVEL (POSITION NUMBER) OF THE OPTION WORD. |
| CLDCLG | | (LOCAL) | METERS | CLOUD CEILING ALTITUDE. (NOT ACTUALLY USED AFTER IT IS SET). THE INDEX OF THE CLOUD CEILING ALTITUDE (ICC) IS THE NUMBER USED TO ACCESS DATA THAT DEPENDS ON CLOUD CEILING ALTITUDE.) |
| CLOUD(6) | | WEATHR | METERS | ARRAY OF CLOUD CEILING ALTITUDE VALUES. |
| CM(10) | | (LOCAL) | | CONTAINS COMMENT LINE GENERATED BY PROGRAM AND READY TO BE PROCESSED BY SUBROUTINE COMMNT. |
| CMENT(13, 20) | | COMENT | | CMENT(I, J) CONTAINS CHARACTERS 10*(I-1)+1 THROUGH 10*I OF THE JTH LINE OF COMMENTS TO BE PRINTED WITH THE CURRENT CASE |
| CODE | | (LOCAL) | | CODE AND ICODE ARE EQUIVALENT. (SEE ICODE) |
| COUT(4, 40) | | (LOCAL) | | ARRAY INTO WHICH THIRD COLUMN OF CASE HEADING OUTPUT BLOCKS ARE ENCODED PRIOR TO BEING PRINTED OUT IN SUBROUTINE HEADER. |
| COMBUF(13) | | COMENT | | COMMENT BUFFER. IF FOR ANY CODE CASE THE NUMBER OF $$ COMMENTS EXCEEDS THE SECOND DIMENSION OF THE CMENT ARRAY (ORIGINALLY SET AT 20), THEN THE COMMENT THAT CAUSES THE OVERFLOW IS STORED IN COMBUF AND AN ERROR PRINT OCCURS FOLLOWED BY A STOP WITH MESSAGE. |
| CRNG(11, 2) | | RNGLOS | NONE, METERS | CUMULATIVE ACQUISITION RANGE DISTRIBUTION. CRNG(I, 1) IS THE PROBABILITY THAT THE ACQUISITION RANGE IS LESS THAN OR EQUAL TO CRNG(I, 2). (CRNGD(I,1) IS UNITLESS; CRNG(I, 2) IS IN METERS). |
| [illegible](70) | | DVALUE | | DEFAULT VALUES FOR PARAMETERS ASSOCIATED WITH KEYWORDS. |
| [illegible]AT | | RUNDAT | | DATE OF RUN FOR CURRENT COMPUTER JOB. (OBTAINED FROM COMPUTER'S INTERNAL CALENDAR.) |

AD-A100 285 ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY ABERDEEN PROV--ETC F/G 19/1
COPPERHEAD OPERATIONAL PERFORMANCE EVALUATION (COPE): COMPUTER --ETC(U)
MAR 81 R S SANDMEYER

UNCLASSIFIED AMSAA-TR-318 NL



AD A
100285

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| DAYLBL(2) | | HEADNG | | ALPHANUMERIC LABELS USED IN LABELING OUTPUT HEADINGS AS DAY OR NIGHT. |
| DDT | | DVALUE | | NOT USED. |
| DECFLG | | (LOCAL) | | IF DECFLG=1.0, A DECIMAL POINT HAS BEEN ENCOUNTERED IN C ARRAY; IF DECFLG=0.0. NO DECIMAL POINT HAS BEEN ENCOUNTERED IN C ARRAY. |
| DECPNT | | SYMBOL | | ALPHANUMERIC SYMBOL FOR DECIMAL POINT STORED IN A1 FORMAT. |
| DEFB | | PEDESC(LOCAL) IN PENAME) | METERS | DEFLECTION BIAS (DISTANCE FROM FOOTPRINT CENTROID TO POINT OF CLOSEST APPROACH OF TARGET). |
| DELTAT | | (LOCAL) | SECONDS | TIME ROUND ARRIVES ON TARGET MINUS TIME LAST TARGET VEHICLE IN COLUMN PASSES POINT OF CLOSEST APPROACH TO FOOTPRINT CENTROID. IF DELTAT AS COMPUTED ABOVE IS NEGATIVE, THEN A VALUE OF ZERO IS USED. |
| DESRNG(3, 2) | | DESRNG | METERS | DESRNG(I,J) IS THE MAXIMUM DESIGNATOR RANGE FOR THE ITH DESIGNATOR TYPE AND THE JTH DAY OR NIGHT CONDITION (J=1 FOR DAY, J=2 FOR NIGHT). |
| DETTIM | | RSPTIM | SECONDS | TIME FROM TARGET UNMASK TO OBSERVER'S CALL FOR COPPERHEAD. |
| DETTMA(10, 2) | | RSPTIM | NONE, SEC | DETECTION TIME CUMULATIVE DISTRIBUTION ARRAY. DETTMA(I,1) IS THE PROBABILITY THAT DETECTION TIME IS LESS THAN OR EQUAL TO DETTMA(I,2). |
| DFCODE | | RECNAM | | RECORD NAME KEY FOR THE DIRECT FIRE DATA RECORD LAST READ FROM THE WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| DFDOKL(10,3) | | DODF | METERS, NONE | DFDOKL(I,1) IS THE ITH RANGE CLASS UPPER LIMIT: DFFOKL(I,2) IS THE PROBABILITY THAT DIRECT FIRE OBSCURES DESIGNATOR'S VIEW OF TARGET AT RANGE DFDOKL(I,1) BUT DOES NOT KILL DESIGNATOR: DFDOKL(I,3) IS THE PROBABILITY THAT DIRECT FIRE KILLS THE DESIGNATOR AT RANGE DFDOKL(I,1). |
| DFSDAT | | (LOCAL) | METERS | ARRAY TO WHICH ELEMENTS OF DIRECT FIRE SUPPRESSION DATA CLOCK ARE EQUIVALENCED. |
| DFSLBL(2,2) | | HEADNG | | ARRAY CONTAINING LABELS FOR DIRECT FIRE SUPPRESSION LEVEL PRINT OUT. |
| DGTLBL(2,2) | | HEADNG | | ARRAY CONTAINING LABELS FOR COMMUNICATIONS PRINT OUT. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| DIDIGT | | DVALUE | | DEFAULT COMMUNICATIONS TYPE (DIGITAL). |
| DIDN | | DVALUE | | DEFAULT DAY OR NIGHT TIME (DAY). |
| DISBVH | | TARGET | METERS | MEAN DISTANCE BETWEEN CONSECUTIVE VEHICLES IN TARGET COLUMN. |
| DISPM(12) | | DISPLY | | DISPM(I) IS THE FIRST THREE LETERS OF THE ITH MONTH NAME USED IN PROGRAM. (A3 FORMAT) |
| DISPT(24) | | DISPLY | | DISPT(I) IS THE FOUR DIGIT TIME OF DAY FOR THE ITH TIME OF DAY PLAYED. (A4 FORMAT) |
| DLTT(6) | | HIT | SECONDS | ARRAY OF DELAY TIME VALUES USED FOR INTERPOLATING IN PROB-ABILITY OF ENGAGEMENT ARRAY. |
| DOARPK | | MISC | | PROBABILITY DESIGNATOR HAS BEEN KILLED BY PREPARATORY ARTILLERY FIRES. |
| DOKILD | LOGICAL | DODF | | FLAG USED TO INDICATE THAT DESIGNATOR HAS BEEN KILLED BY DIRECT FIRE. (TRUE = KILLED; FALSE = NOT KILLED) |
| DOLBL(3,2) | | HEADNG | | ARRAY OF DESIGNATOR LOCATION LABELS USED FOR PRINT OUT. |
| DOLLAR | | SYMBOL | | DOLLAR SIGN SYMBOL IN A1 FORMAT. |
| DPRT | | DVALUE | SECONDS | DEFAULT VALUE FOR PARAMETERIZED RESPONSE TIME. |
| DSMK(2) | | DVALUE | | DEFAULT NUMBER OF SMOKE ROUNDS FIRED BY RED. DSMK(1) FOR TYPE 1 SMOKE ROUNDS: DSMK(2) FOR TYPE 2 SMOKE ROUNDS. |
| DTRNG | | RANGE | METERS | DESIGNATOR-TO-TARGET RANGE. |
| DUM1 | | RSPTIM | | DUMMY VARIABLE NO LONGER USED (BUT RETAINED TO AVOID CHANGING COMMON BLOCK SIZE). |
| DURLOS | | TIME | SECONDS | DURATION OF LINE-OF-SIGHT (TIME FROM WHEN FIRST VEHICLE IN COLUMN UNMASKS UNTIL LAST VEHICLE IN COLUMN LEAVES LINE-OF-SIGHT SEGMENT). |
| DVALUE(70) | | (LOCAL) | | DVALUE(I) IS EQUIVALENT TO D(I). |
| EDDTIM | | (LOCAL) | SECONDS | TIME TO ENTER FIRE REQUEST ON DIGITAL MESSAGE DEVICE. |
| ENDFLG | LOGICAL | FLAG | | FLAG INDICATING NO FURTHER CASES TO BE READ. |
| ENDREP | LOGICAL | FLAG | | FLAG INDICATING REPLICATION (OR FIRING OF ROUND) IS TO BE ENDED. |

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| FDT | | (LOCAL) | | USED TO CALCULATE FDT2. |
| FDT1<br>FDT2 | | (LOCAL) | | USED FOR INTERPOLATING WITH RESPECT TO DELAY TIME IN THE PROBABILITY OF ENGAGEMENT TABLES. |
| FIRSTL | LOGICAL | LOGFLG | | FLAG USED TO INDICATE FIRST INPUT LINE IN CHECKING FOR SEQUENCE NUMBERS. |
| FPRCNT | | FUNCTION | | FUNCTION VALUE, SPECIAL FORMATTING OF PERCENTAGES USED FOR PRINT OUT. |
| FRAC | | (LOCAL) | | USED TO INTERPOLATE WITH RESPECT TO RANGE IN THE DFDOKL ARRAY. |
| FRONT | | SMOKED | KM | WIDTH OF FRONT WHICH RED WANTS TO SMOKE. |
| FSRT(14,2) | | FSRESP | NONE,<br>SECONDS | FT SILL'S CUMULATIVE RESPONSE TIME DISTRIBUTION. FSRT(I,1) IS THE PROBABILITY THAT THE TOTAL DESIGNATOR-FDC-BATTERY RESPONSE TIME IS LESS THAN OR EQUAL TO FSRT(I,2). |
| FSRTIM | | (LOCAL) | SECONDS | TOTAL DESIGNATOR-FDC-BATTERY RESPONE TIME AS SAMPLED FROM FSRT ARRAY WHEN USING FT SILL RESPONSE TIME DISTRIBUTION. |
| FTSILL | LOGICAL | LOGFLG | | FLAG TO INDICATE WHETHER FT SILL RESPONSE TIME DISTRIBUTION IS BEING USED. |
| GAMMA | | FUNCTION | | FUNCTION VALUE. GAMMA DISTRIBUTED RANDOM VARIABLE. |
| GTR | | (F.P.) | KM | GUN-TO-TARGET RANGE AS USED IN SUBROUTINE PENAME. |
| GTRNG | | PEDESC | KM | GUN-TO-TARGET RANGE. |
| HUMID(2) | | WEATHR | | HUMID(I) IS THE PROBABILITY THAT THE RELATIVE HUMIDITY FALLS IN CLASS I. (IF I=1, RELATIVE HUMIDITY $\leq$65%; IF I=2, RELATIVE HUMIDITY >65%). |
| I | | (LOCAL)<br>(F.P.) | | USED AS CONTROL VARIABLE IN VARIOUS DO-LOOPS AND IMPLIED DO-LOOPS. ALSO USED AS A FORMAL PARAMETER (RANDOM NUMBER SEED AND SUBSCRIPT) AND A COUNTER. |
| IABRT | | (LOCAL)<br>(F.P.) | | INDEX NUMBER OF CAUSE OF MISSION (OR ROUND) ABORT. |
| IACQRL | | (LOCAL) | | LENGTH OF ACQUISITION DATA RECORD ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| IATENG | | (LOCAL) | | NUMBER OF ATTEMPTED ENGAGEMENTS FOR THE CURRENT CASE. |
| IB | | (LOCAL & F.P.) | | USED AS SECOND SUBSCRIPT OF B ARRAY IN NUMRIC AND FINUIT. |
| IBAIL | | BAIL | | FLAG INDICATING WHICH BAIL-OUT CHECK IS BEING MADE. IBAIL=1 INDICATES PRE-COMMUNICATIONS CHECK; IBAIL=2 INDICATES POST-COMMUNICATIONS CHECK. |
| IC | | (LOCAL) | | CONTROL VARIABLE FOR IMPLIED DO-LOOP IN DECODE STATEMENT. ALSO AN UNUSED RETURNED PARAMETER FROM A CALL TO SMPLCD. |
| ICC | | WEATHR | | ICC INDICATES WHICH OF THE SIX CLOUD CEILING ALTITUDES IS BEING USED ON THIS REPLICATION. IF CLOUD FREE LINE-OF-SIGHT EXISTS, ICC=1 (HIGHEST CEILING) IS USED. |
| ICCPFG | | RUNDAT | | CALCOMP PLOT FLAG. ICCPFG=0 INDICATES NO CALCOMP PLOT HAS BEEN REQUESTED FOR THE CURRENT CASE: ICCPFG=1 INDICATES THAT CALCOMP PLOT OUTPUT HAS BEEN REQUESTED. |
| ICFLOS | | (LOCAL) | | CLOUD FREE LINE-OF-SIGHT FLAG. ICFLOS=1 INDICATES A CLOUD FREE LINE-OF-SIGHT; ICFLOS=2 INDICATES NO CLOUD FREE LINE-OF-SIGHT. |
| ICHNL | | (LOCAL) | | NUMBER OF OPTION WORDS IN CHARA ARRAY TO BE CHECKED IN SEARCH FOR MATCH WITH OPTION WORDS OF CURRENT KEYWORD. |
| ICHRRL | | (LOCAL) | | NUMBER OF WORDS IN OPTION WORD DATA RECORD ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| ICLASS | | (LOCAL) (F.P.) | | WHEN A CUMULATIVE DISTRIBUTION IS SAMPLED USING THE SMPLCD SUBROUTINE, THE VALUE RETURNED IN ICLASS IS THE NUMBER OF THE INTERPOLATION INTERVAL FROM WHICH THE SAMPLE WAS OBTAINED. |
| ICLNGF | | (LOCAL) | | CLOUD CEILING FLAG. ICLNGF=1 INDICATES A CLOUD CEILING; ICLNGF=2 INDICATES NO CLOUD CEILING (I.E., SCATTERED CLOUDS) |
| ICMFLG | | (LOCAL) | | COMMENT FLAG. ICMFLG=0 INDICATES NO TIME OF DAY/WEATHER INCONSISTENCY COMMENT WILL BE PRINTED FOR THIS CASE; ICMFLG=1 INDICATES SUCH A COMMENT WILL BE PRINTED. |
| ICMNT | | COMENT | | NUMBER OF $$ COMMENT LINES TO BE PRINTED FOR THIS CASE. |
| ICODE | | ICODE | | RECORD NAME KEY WHEN READING FROM WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). ICODE IS EQUIVALENT TO CODE. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| ICRD | | | | INDEX NUMBER USED TO CHECK WHETHER MORE THAN ONE INPUT LINE HAS BEEN READ WITH THE SAME KEYWORD FOR EACH CASE. |
| IDABRT(3,19) | | ABRLBL | | ALPHANUMERIC DESCRIPTORS OF CAUSES OF MISSION (OR ROUND) ABORTS. IDABRT(I,J) CONTAINS CHARACERS 10*(I-1)+1 THROUGH 10*I OF ABORT DESCRIPTOR NUMBER J. |
| IDCHAR | | FUNCTION | | FUNCTION VALUE. A NUMERICAL VALUE USED TO INDICATE TO WHICH CLASS OF SYMBOLS A GIVEN CHARACTER BELONGS. |
| IDCHR | | (LOCAL) | | A NUMERICAL VALUE INDICATING TO WHICH CLASS OF SYMBOLS A PARTICULAR CHARACTER BELONGS. |
| IDCODE | | (F.P.) | | RECORD NAME KEY FOR A PROBABILITY OF ENGAGEMENT DATA RECORD TO BE READ OR INTERPRETED. |
| IDELTT | | (LOCAL) | | SUBSCRIPT OF INDEX ARRAY USED WHEN DELTAT IS ZERO. |
| IDELT1 | | (LOCAL) | | SUBSCRIPT OF INDEX ARRAY USED FOR INTERPOLATION WHEN DELTAT IS POSITIVE. |
| IDELT2 | | (LOCAL) | | SUBSCRIPT OF INDEX ARRAY USED FOR INTERPOLATION WHEN DELTAT IS POSITIVE (IDELT2=IDELT1+1). |
| IDFL | | (LOCAL) | | IDFL=1 INDICATES HIGH LEVEL OF DIRECT FIRE SUPPRESSION; IDFL=2 INDICATES NO DIRECT FIRE SUPPRESSION. (IDFL IS USED TO PRINT CASE HEADING LABELS.) |
| IDFSRL | | (LOCAL) | | DIRECT FIRE SUPPRESSION DATA RECORD LENGTH ON WORD ADDRES-SABLE MASS STORAGE FILE (TAPE 11). |
| IDIGTL | | FLAG | | FLAG TO INDICATE WHICH METHOD OF COMMUNICATION IS TO BE FIRST CHOICE WHEN DESIGNATOR CALLS REQUEST FOR FIRE TO FDC. (IDIGTL=1 INDICATES DIGITAL COMMUNICATION; IDIGTL=2 INDICATES VOICE COMMUNICATION.) |
| IDN | | FLAG | | FLAG TO INDICATE DAY OR NIGHT. (IDN=1 INDICATES DAYTIME; IDN=2 INDICATES NIGHTTIME.) |
| IDOLOC | | (LOCAL) | | FLAG TO INDICATE DESIGNATOR OPERATOR LOCATION. (IDOLOC=1 INDICATES VANTAGE POINT LOCATION; IDOLOC=2 INDICATES MANEUVER UNIT LOCATION). |
| IDSGTP | | RSPTIM | | FLAG TO INDICATE DESIGNATOR TYPE. (IDSGTP=1 INDICATES GLLD, IDSGTP=2 INDICATES MULE. AND IDSGTP=3 INDICATES LTD). |

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| IDT | | (F.P.) | | FORMAL PARAMETER INDICATING DESIGNATOR TYPE. VALUES OF 1,2, AND 3 HAVE SAME MEANING AS FOR THE VARIABLE IDSGTP. |
| IGO | | (LOCAL) | | COMPUTED GO TO PARAMETER USED TO SEND CONTROL TO SECTION OF SUBROUTINE INPUT THAT HANDLES A PARTICULAR CLASS OF INPUT LINES. |
| IGTR | | (LOCAL) | KM | GUN-TO-TARGET RANGE ROUNDED TO NEAREST INTEGER. |
| II | | (LOCAL) | | USED AS SUBSCRIPT FOR INDEXING PETBL ARRAY. ALSO USED AS DO-LOOP CONTROL VARIABLE. |
| II1 | | (LOCAL) | | USED AS SUBSCRIPT FOR INDEXING PETBL ARRAY. |
| II2 | | (LOCAL) | | USED AS SUBSCRIPT FOR INDEXING PETBL ARRAY. |
| IJ | | (LOCAL) | | CONTROL VARIABLE IN VARIOUS DO-LOOPS. |
| IK | | (LOCAL) | | CONTROL VARIABLE IN DO-LOOP. |
| IKTEST | | (LOCAL) | | DO-LOOP CONTROL VARIABLE IN REINITIALIZING KTEST ARRAY. |
| IL | | (LOCAL) | | CONTROL VARIABLE IN IMPLICIT DO-LOOP. |
| IMATCH | | (LOCAL) | | SUBSCRIPT OF KEYWORD CHARACTER STRING FOUND TO MATCH A CURRENT INPUT LINE CHARACTER STRING. |
| IMATCH1 | | (LOCAL) | | STORES SUBSCRIPT OF KEYWORD CHARACTER STRING MATCHING FIRST CURRENT INPUT LINE KEYWORD WHILE SECOND KEYWORD IS BEING CHECKED FOR A MATCH (APPLIES ONLY TO "TEMPORARY", "OVERRIDE" AND "RESET" KEYWORDS). |
| IMNTH | | (LOCAL) | | INDEX NUMBER OF MONTH OF CURRENT CASE'S WEATHER DATA. (E.G., IF JUNE AND DECEMBER ARE THE ONLY MONTHS PLAYED, IMNTH=1 FOR JUNE AND IMNTH=2 FOR DECEMBER). |
| IMUTS | | TARGET | | FLAG AND SUBSCRIPT INDICATING WHETHER TARGET IS PAST THE POINT OF CLOSEST APPROACH TO THE CENTROID OF THE FOOTPRINT. (IMUTS=1 IF TARGET IS PAST POINT OF CLOSEST APPROACH; IMUTS=2 IF NOT). |
| IMXNR | | (LOCAL) | | NUMBER OF OPTION WORDS TO BE CHECKED FOR MATCH ON CURRENT INPUT LINE. |
| IN | | (LOCAL) | | NUMBER OF GAMMA DISTRIBUTION RANDOM DEVIATES TO BE RETURNED BY CALL OF IMSL SUBROUTINE GGAMA. |

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| INDEX(6,5,2) | | HIT | | INDEXING ARRAY FOR REFERENCING PETBL ARRAY. INDEX(I,J,K) IS THE NUMBER OF THE PETBL TO BE USED FOR CLOUD CEILING I, VISIBILITY RANGE J, AND IMUTS VALUE K. |
| INDX11(2001) | | INDEX1 | | ARRAY CONTAINING MASTER INDEX FOR WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| INVDRL | | (LOCAL) | | INVARIANT DATA RECORD LENGTH ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| IOCSN | | (LOCAL) | | NUMBER OF OCCASIONS FOR THE CURRENT CASE (EQUAL TO NREP). |
| IOPTNM(70) | | (LOCAL) | | IOPTNM(I) IS THE NUMBER OF DIFFERENT OPTION WORD CHARACTER STRINGS ALLOWED FOR KEYWORD I. |
| IOVER(70) | | OVER | | IOVER(I)=1 INDICATES "OVERRIDE" OPTION IN EFFECT FOR KEY-WORD I; IOVER(I)=2 INDICATES "TEMPORARY" OPTION IN EFFECT FOR KEYWORD I. |
| IPERL | | (LOCAL) | | PROBABILITY OF ENGAGEMENT DATA RECORD LENGTH ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| IPOINT(70) | | POINT | | IPOINT(I) IS THE VALUE OF THE COMPUTED GO TO PARAMETER THAT WILL SEND CONTROL THE PROPER PART OF SUBROUTINE INPUT FOR HANDLING KEYWORD I INPUTS. |
| IPPFG | | RUNDAT | | PRINTER PLOT FLAG. IPPFG=1 INDICATES PRINTER PLOT OUTPUT REQUESTED FOR THIS CASE; IPPFG=0 INDICATES PRINTER PLOT OUTPUT NOT REQUESTED FOR THIS CASE. |
| IPREPL | | FLAG | | PRE-PLANNED TARGET FLAG. IPREPL=1 INDICATES PRE-PLANNED TARGET; IPREPL=2 INDICATES TARGET OF OPPORTUNITY. |
| IPRINT | | (LOCAL) | | FLAG TO INDICATE WHETHER CURRENT RANDOM NUMBER SEEDS HAVE YET BEEN WRITTEN TO TAPE 8, IPRINT=0 INDICATES THEY HAVE NOT; IPRINT=1 INDICATES THEY HAVE. |
| IPRTY | | FLAG | | PRIORITY FLAG. IPRTY=2 INDICATES PRIORITY PRE-PLANNED TARGET; IPRTY=0 INDICATES NO SPECIAL PRIORITY. |
| IRAY(6) | | (LOCAL) | | ARRAY CONTAINING ERROR PROCESSING SPECIFICATIONS FOR CALL OF SYSTEMC SUBROUTINE FOR NON-STANDARD RECOVERY FOR CDC FORTRAN ERROR 104. |
| IRC | | (LOCAL) | | UNUSED VALUF RETURNED BY A CALL OF SUBROUTINE SMPLCD. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| IRF | | (LOCAL) | | INITIAL PARAMETER OF A DO-LOOP (IRF=1). |
| IRM | | (LOCAL) | | SUBSCRIPT USED IN INTERPOLATING WITH RESPECT TO RANGE IN RNGTTF ARRAY TO OBTAIN PROBABILITY OF HIT GIVEN AN ENGAGEMENT. |
| IRN | | (LOCAL) (F.P.) | | SUBSCRIPT USED IN INTERPOLATING WITH RESPECT TO RANGE IN PSTTBL ARRAY TO OBTAIN PROBABILITIES OF THE VARIOUS TARGET POSTURES. ALSO FORMAL PARAMETER FOR RANDOM NUMBER SEED IN GAMMA. |
| IRNDM | | (F.P.) | | RANDOM NUMBER SEED TO BE USED WHEN CALLING URAN31 FROM SMPLCD. |
| IRNDOC | | FLAG | | RANDOM OCCURRENCE FLAG. IRNDOC=1 INDICATES RANDOM OCCURRENCE LINE-OF-SIGHT METHODOLOGY USED FOR THIS CASE; IRNDOC=0 INDICATES SHOOTING GALLERY LOS METHODOLOGY USED FOR THIS CASE. |
| IRNG1 | | (LOCAL) | KM | LOWER LIMIT OF RANGE BRACKET USED TO INTERPOLATE IN PETBL ARRAY. |
| IRNG2 | | (LOCAL) | KM | UPPER LIMIT OF RANGE BRACKET USED TO INTERPOLATE IN PETBL ARRAY. |
| IRORL | | (LOCAL) | | RANDOM OCCURRENCE DATA RECORD LENGTH ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| IROW | | (LOCAL) | | NUMBER OF THE ROW OF THE SEGLOS ARRAY TO BE USED IN SAMPLING FOR LINE-OF-SIGHT SEGMENT LENGTH ON THIS REPLICATION. |
| IRSPRL | | (LOCAL) | | RESPONSE TIME DATA RECORD LENGTH ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| IR(28) | | (LOCAL) | | ARRAY OF RANDOM NUMBER SEEDS. IR(1) IS EQUIVALENT TO IR1, IR(2) TO IR2, . . . ., IR(28) TO IR28. |
| IR1, IR2, . . ., IR28 | | RANDOM | | RANDOM NUMBER SEEDS. IR1 IS SEED FOR FIRST RANDOM NUMBER STREAM, IR2 FOR SECOND, ETC. |
| ISHOT | | (LOCAL) | | NUMBER OF REPLICATIONS OF CURRENT CASE THAT REACHED THE STAGE IN THE MISSION WHERE A ROUND WAS FIRED. |
| ISTART | | (LOCAL) | | DO-LOOP INITIAL PARAMETER. ISTART IS THE SUBSCRIPT FOR THE FIRST LINE TO BE READ FROM CARDS AS RESULT OF A "TEMPORARY" OPTION FOR PEDATA. |

GLOSSARY- CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| IT(10) | | (LOCAL) | | IT(1) = NUMBER OF CURRENT INPUT LINE'S KEYWORD. IT(J) IS THE NUMBER OF THE (J-1)ST OPTION WORD CHOICE OF THE CURRENT LINE FOR J>1. |
| ITDGTL | | (LOCAL) | | FLAG FOR CURRENT COMMUNICATIONS MODE. ITDGTL=1 FOR DIGITAL; ITDGTL=2 FOR VOICE. |
| ITGTPS | | TARGET | | TARGET POSTURE FLAG. ITGTPS=1 FOR FULLY EXPOSED TARGET; ITGTPS=2 FOR HULL DEFILADE TARGET. |
| ITGTTP | | TARGET | | TARGET TYPE NUMBER. ITGTTP=1 FOR TARGET TYPE 1, ITGTTP=2 FOR TARGET TYPE 2, ETC. |
| ITIMDL | | (LOCAL) | | ITIMDL=1 INDICATES FIRST METHOD OF MODELING RESPONSE TIME IS PLAYED (SINGLE DISTRIBUTION SAMPLED); ITIMDL=2 INDICATES SECOND METHOD OF MODELING RESPONSE TIME IS PLAYED (SINGLE VALUE USED); ITIMDL=3 INDICATES THIRD METHOD OF MODELING RESPONSE TIME IS PLAYED (ADDING CONTRIBUTIONS OF INDIVIDUAL DELAY COMPONENTS). |
| ITPSRL | | (LOCAL) | | TARGET POSTURE DATA RECORD LENGTH FOR WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| ITRRN | | (LOCAL) | | SUBSCRIPT INDICATING TERRAIN TYPE USED FOR CURRENT CASE. USED TO REFERENCE ACQLBL ARRAY FOR OUTPUT HEADING. |
| ITRY | | (LOCAL) | | FLAG USED TO INDICATE WHETHER CURRENT ATTEMPT TO MATCH KEY-WORD IS FOR FIRST OR SECOND KEYWORD OF CURRENT INPUT LINE. APPLIES ONLY TO "TEMPORARY", "OVERRIDE", AND "RESET" FIRST KEYWORDS. |
| ITVEL | | (LOCAL) | METERS/SEC | TARGET VELOCITY. |
| ITYPE | | (LOCAL) | | COMPUTED GO TO PARAMETER USED TO SEND CONTROL TO PROPER READ STATEMENTS WHEN "TEMPORARY" OPTION IS USED. |
| IT1 | | (LOCAL) | | NUMBER OF CURRENT INPUT LINE'S FIRST KEYWORD. |
| IT2 | | (LOCAL) | | NUMBER OF CURRENT INPUT LINE'S SECOND KEYWORD OR FIRST OPTION WORD. |
| IT3 | | (LOCAL) | | NUMBER OF CURRENT INPUT LINE'S SECOND OPTION WORD. |
| IT3FLG | | FLAG | | FLAG USED TO INDICATE WHETHER RUMC IS TO BE CONSTANT. IF IT3FLG=0, RUMC VARIES FROM REPLICATION TO REPLICATION; IF IT3FLG=1, RUMC IS CONSTANT. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| IUNIT | | (F.P.)<br>(LOCAL) | | LOGICAL UNIT NUMBER TO WHICH OUTPUT OF SUBROUTINE ECHO IS TO BE WRITTEN. ALSO UNIT FROM WHICH "TEMPORARY" OPTION PEDATA IS TO BE READ IN SUBROUTINE CREAD. |
| IVL | | WEATHR | KM | VISIBILITY RANGE LIMIT FOR THIS REPLICATION. |
| IVLA | | (LOCAL) | | VISIBILITY RANGE SUBSCRIPT VALUE FOR REFERENCING PETBL ARRAY (IVLA=IVL EXCEPT THAT IF IVL>10, THEN IVLA=10). |
| IW | | (LOCAL) | | COUNTER INDICATING POSITION NUMBER OF THE KEYWORD OR OPTION WORD CURRENTLY BEING ANALYZED IN THE SEPREC SUBROUTINE. |
| IWTHRL | | (LOCAL) | | WEATHER DATA RECORD LENGTH ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| IX | | (LOCAL) | | COUNTER USED IN SEPREC SUBROUTINE SEPARATION ALGORITHM. |
| IXXX | | (LOCAL) | | IXXX=1 INDICATES GOOD WEATHER (CURRENTLY JUNE AND SEPTEMBER) IXXX=2 INDICATES BAD WEATHER (CURRENTLY MARCH AND DECEMBER) IXXX GOVERNS THE VALUE OF THE VELTBL ARRAY TO BE USED IN THE CURRENT CASE. (APPLICABLE ONLY IF TARGET VELOCITY DEFAULT IS IN EFFECT.) |
| I1 | | (LOCAL) | | CONTROL PARAMETER FOR IMPLIED DO-LOOPS. |
| I2 | | (LOCAL) | | CONTROL PARAMETER FOR IMPLIED DO-LOOPS. ALSO RECEIVING ARGUMENT FOR DESIGNATOR TYPE IN CALL OF PEIDNT. |
| J | | (LOCAL) | | CONTROL VARIABLE IN VARIOUS DO-LOOPS. ALSO USED IN CALCULATION OF RANDOM NUMBER IN URAN31 AND AS A COUNTER IN SEPREC. |
| JEND | | (LOCAL) | | DO-LOOP MAXIMUM LIMIT (NUMBER OF DIFFERENT WIND SPEEDS USED IN SMOKE CALCULATION FOR THIS PASQUILL CATEGORY). |
| JJ | | (LOCAL) | | DO-LOOP CONTROL VARIABLE. |
| JK | | (LOCAL) | | CONTROL VARIABLE FOR DO-LOOPS AND IMPLIED DO-LOOPS. |
| JKTEST | | (LOCAL) | | CONTROL VARIABLE FOR DO-LOOP. |
| JL | | (LOCAL) | | CONTROL VARIABLE FOR IMPLICIT DO-LOOP. |
| K | | (LOCAL) | | CONTROL VARIABLE FOR VARIOUS DO-LOOPS AND IMPLICIT DO-LOOPS. ALSO USED IN CALCULATING INITIAL RANDOM NUMBER SEEDS. |

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| KCHAR | | (LOCAL) | | NUMBER OF CHARACTERS IN KEYWORD OR OPTION WORD ABOUT TO BE ENCODED. |
| KI | | (LOCAL) | | POSITION NUMBER OF CHARACTER OF OPTION WORD CURRENTLY BEING CHECKED TO DETERMINE WHETHER OPTION WORD IS A NUMERIC VALUE. |
| KREP | | RPLCTN | | NUMBER OF CURRENT REPLICATION. |
| KRF | | RPLCTN | | NUMBER OF THE ROUND (SHOT) CURRENTLY BEING SIMULATED. |
| KTEST(20,6) | | KTEST | | KTEST(I,J) IS THE NUMBER OF TIMES ABORT CONDITION I IS TESTED FOR ROUND (SHOT) NUMBER J. IF I ≤ MSABLM, THEN KTEST(I,J)=0 FOR J>1; IN SUCH CASES KTEST (I,1) IS THE NUMBER OF TIMES MISSION ABORT CONDITION I IS TESTED. |
| LIMU | | (LOCAL) | | MAXIMUM LIMIT ON DO-LOOP COMPARING KTEST AND NTEST ARRAYS. (LIMU=1 IF COMPARING MISSION ABORT TESTS; LIMU=6 IF COMPARING ROUND ABORT TESTS). |
| LSTCMA | | (LOCAL) | | GIVES POSITION OF LAST COMMA ENCOUNTERED THUS FAR IN SEPARATING KEYWORDS AND OPTION WORDS OF CURRENT INPUT LINE. |
| M | | (LOCAL) | | USED IN CALCULATION OF INITIAL RANDOM NUMBER SEEDS. |
| MSABLM | | ABORT | | MISSION ABORT LIMIT. CAUSE OF ABORT NUMBERS LESS THAN OR EQUAL TO MSABLM ARE MISSION ABORTS WHEREAS ABORT CAUSES NUMBERED GREATER THAN MSABLM ARE ROUND ABORTS. A MISSION ABORT CANCELS ALL FURTHER STEPS IN THE POTENTIAL COPPERHEAD FIRE MISSION (REPLICATION) WHEREAS A ROUND ABORT MEANS ONLY THAT THE CURRENT ROUND IS UNSUCCESSFUL. |
| MXDRNG | REAL | RANGE | METERS | MAXIMUM DESIGNATOR-TO-TARGET RANGE AT WHICH DESIGNATOR OPERATOR WILL CALL FOR COPPERHEAD FIRE. |
| N | | (F.P.) | | NUMBER OF CHARACTERS IN INPUT LINE WHOSE ITH CHARACTER IS CURRENTLY BEING CHECKED FOR SYMBOL CLASS. |
| NABORT(20,6) | | ABORT | | NABORT(I,J) IS THE NUMBER OF TIMES TEST I WAS FAILED ON ROUND (SHOT) J. IF I≤ MSABLM, THEN NABORT(I,J)=0 FOR J>1; IN SUCH CASES NABORT(I,1) IS THE NUMBER OF TIMES MISSION ABORT TEST I WAS FAILED. |

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| NC(80) | | (LOCAL) | | NC(I) IS THE NUMBER INDICATING WHICH TYPE OF SYMBOL THE ITH CHARACTER OF THE CURRENT INPUT LINE IS. (0 INDICATES NUMERAL OR MINUS SIGN, 1 INDICATES LETTER OR APOSTROPHE, 2 INDICATES DECIMAL POINT, 3 INDICATES BLANK, 4 INDICATES SEPARATOR, 5 INDICATES A DOLLAR SIGN, AND 6 INDICATES ANY OTHER CHARACTER.) |
| NCASE | | RUNDAT | | SEQUENCE NUMBER OF CASE CURRENTLY BEING EXECUTED IN THIS PROGRAM RUN. |
| NCC | | WEATHR | | NUMBER OF DIFFERENT CLOUD CEILING ALTITUDE VALUES USED IN WEATHER DATA. |
| NCODE | | (LOCAL) | | USED IN CREATING AND DECIPERING PEDATA RECORD NAME. |
| NCPR(10) | | (LOCAL) | | NCPR(I) IS THE NUMBER OF CHARACTERS IN THE ITH LINE OF THE B ARRAY (ITH FIELD ON INPUT LINE AFTER COMPRESSING MOST BLANKS AND POSSIBLY ADDING DECIMAL POINTS). |
| NCRD(70) | | (LOCAL) | | NCRD(I) IS THE NUMBER OF INPUT LINES READ FOR THE CURRENT CASE THAT DETERMINE THE OPTIONS USED WITH KEYWORD NUMBER I. |
| NDFSP | | DODF | | NUMBER OF RANGE POINTS USED IN DIRECT FIRE SUPPRESSION DISTRIBUTION (DFDOKL ARRAY). |
| NDOKIL | | DODF | | NUMBER OF REPLICATIONS IN WHICH DESIGNATOR OPERATOR HAS BEEN KILLED FOR CURRENT CASE. |
| NDR | | (F.P.) | | INDICATES NUMBER OF POINTS USED IN DEFINITION OF CUMULATIVE DISTRIBUTION FUNCTION NOW BEING SAMPLED. |
| NDRM1 | | (LOCAL) | | MAXIMUM LIMIT ON DO-LOOP. (EQUALS NDR-1). |
| NDT | | RSPTIM | | NUMBER OF POINTS USED IN DETECTION TIME DISTRIBUTION. |
| ND1 | | (F.P.) | | FIRST DIMENSION OF ARRAY WHOSE ELEMENTS DEFINE CUMULATIVE DISTRIBUTION BEING SAMPLED. |
| ND2 | | (F.P.) | | SECOND DIMENSION OF ARRAY WHOSE ELEMENTS DEFINE CUMULATIVE DISTRIBUTION BEING SAMPLED. |
| NKILL(6) | | (LOCAL) | | NKILL(I) IS THE NUMBER OF REPLICATIONS IN WHICH THE ITH ROUND OF THE FIRE MISSION KILLED A VEHICLE FOR THE CURRENT CASE. |
| NLRTC | | (LOCAL) | | NUMBER OF INPUT LINES READ SO FAR FOR THE CURRENT CASE. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| NP(9) | | (LOCAL) | | USED TO CONSTRUCT THE PEDATA RECORD NAME. NP(I) IS THE INDEX NUMBER (IN THE XVALUE ARRAY) OF THE CURRENT VALUE OF THE ITH PARAMETER OF THE PEDATA RECORD NAME. |
| NPLOSR | | RANDOC | | NUMBER OF RANGE POINTS USED IN THE PROBABILITY OF LINE-OF-SIGHT FOR RANDOM OCCURRENCE (RNGPLS ARRAY). |
| NPP | | RNGLOS | | NUMBER OF POINTS USED IN DEFINING CUMULATIVE DISTRIBUTIONS FOR LINE-OF-SIGHT SEGMENT LENGTHS. |
| NRE | | (LOCAL) | | NUMER OF KEYWORDS AND OPTION WORDS ENCOUNTERED ON CURRENT INPUT LINE. |
| NREC | | ACHAR | | NUMBER OF DIFFERENT KEYWORD CHARACTER STRINGS ALLOWED AS INPUTS. |
| NREP | | RPLCTN | | NUMBER OF REPLICATIONS TO BE DONE FOR THE CURRENT CASE. |
| NRF | | RPLCTN | | NUMBER OF ROUNDS (SHOTS) TO BE FIRED FOR EACH REPLICATION (POTENTIAL FIRE MISSION SAMPLE) FOR THIS CASE. |
| NRNGCB | | (LOCAL) | | NUMBER OF ROWS TO BE READ INTO SEGLOS DISTRIBUTION. |
| NRNGCL | | RNGLOS | | NUMBER OF RANGE CLASSES INTO WHICH LINE-OF-SIGHT SEGMENT LENGTH DATA IS DIVIDED. |
| NRNGPT | | (LOCAL) | | MAXIMUM LIMIT ON DO-LOOP. (EQUALS NRNGPS) |
| NRNGPS | | HIT | | NUMBER OF RANGE POINTS FOR WHICH TARGET POSTURE DATA IS ENTERED IN RNGPST ARRAY. |
| NRNGTT | | HIT | | NUMBER OF RANGE POINTS FOR WHICH "TRUE TARGET FACTOR" DATA IS ENTERED IN RNGTTF ARRAY. |
| NRP | | RNGLOS | | NUMBER OF POINTS USED IN CUMULATIVE ACQUISITION RANGE DISTRIBUTION TABLE (CRNGD ARRAY). |
| NRS | | (F.P.) | | ROW OF A ARRAY IN WHICH INTERPOLATION IS TO BE PERFORMED TO OBTAIN RANDOM VARIABLE VALUE IN SUBROUTINE SMPLCD. |
| NSMK2 | | SMOKED | TYPE 1 SMOKE ROUNDS | NUMBER OF TYPE 1 SMOKE ROUNDS FIRED BY RED TO CREATE SMOKE SCREEN. |
| NSMK5 | | SMOKED | TYPE 2 SMOKE ROUNDS | NUMBER OF TYPE 2 SMOKE ROUNDS FIRED BY RED TO CREATE SMOKE SCREEN. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| NTEST(20,6) | | (LOCAL) | | NTEST(I,J) IS THE NUMBER OF TIMES ABORT CONDITION I HAS BEEN TESTED FOR ROUND J FOR THIS CASE. FOR I≤ MSABLM, NTEST(I,J)= NTEST(J,1) FOR J>1. |
| NTRC | | COMENT | | NUMBER OF "TRIPLE $" COMMENTS READ AND WRITTEN WITH CURRENT CASE. |
| NUMRIC | LOGICAL | FUNCTION VALUE | | TRUE, IF THE ITEM TESTED IS NUMERIC (CONTAINS ONLY NUMERALS, "-", AND "." AS SYMBOLS); FALSE, IF ITEM IS NOT NUMERIC. |
| NVEHCL | | TARGET | | NUMBER OF VEHICLES IN TARGET UNIT. |
| NVEHKL | | RPLCTN | | NUMBER OF VEHICLES KILLED BY COPPERHEAD THUS FAR IN CURRENT REPLICATION. (APPLIES ONLY TO RANDOM OCCURRENCE LINE-OF-SIGHT OPTION). |
| NVL | | WEATHR | | NUMBER OF DIFFERENT METEOROLOGICAL VISIBILITY RANGE LIMITS USED WITH WEATHER DATA. |
| NXTLIN | | (LOCAL) | | COUNTER USED IN KEEPING TRACK OF LINE NUMBERS IN PRINTING OF CASE HEADING OUTPUT. |
| N1 | | (F.P.) | | VARIABLE ARRAY DIMENSION. |
| N2 | | (F.P.) | | VARIABLE ARRAY DIMENSION. |
| N3 | | (F.P.) | | VARIABLE ARRAY DIMENSION. |
| OUT(6) | | (LOCAL) | | ARRAY USED TO STORE ITEMS ABOUT TO BE WRITTEN ON THE PRINTOUT. |
| PAREN | LOGICAL | (F.P.) | | FLAG TO INDICATE WHETHER A RIGHT PARENTHESIS IS TO BE INCLUDED IN FORMATTING A PERCENTAGE. (TRUE INDICATES PARENTHESIS IS TO BE INCLUDED; FALSE INDICATES IT IS NOT). |
| PASPCT(20,6) | | (LOCAL) | | PASPCT(I,J) IS THE PERCENTAGE OF REPLICATIONS TESTED AT ABORT CONDITION I THAT PASSED (DID NOT ABORT) FOR ROUND J. (IF I< MSABLM, PASPCT(I,J) = PASPCT(I,1) FOR J>1; IF NTEST(I,J)=0, PASPCT(I,J) IS SET TO -1). |
| PASQL(6) | | WEATHR | | PASQL(I) IS THE PROBABILITY OF OCCURRENCE OF PASQUILL ATMOSPHERIC STABILITY CATEGORY I FOR THE CURRENT CASE'S WEATHER CONDITIONS. |
| PASQT | | (LOCAL) | | THE PASQL VALUES ARE GROUPED IN PAIRS. PASQT IS THE SUM OF PASQL(2*I-1) AND PASQL(2*I) FOR THE CURRENT VALUE OF I. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| PB(80,10) | | (LOCAL) | | PB(I,J) IS THE ITH CHARACTER OF THE JTH KEYWORD OR OPTION WORD OF THE CURRENT INPUT LINE. |
| PCRMSG | | RSPTIM | | PROBABILITY THAT THE DESIGNATOR-TO-FDC CALL-FOR-FIRE MESSAGE CONTAINS THE CORRECT FIRE MISSION INFORMATION AND IS CORRECTLY INTERPRETED. |
| PDOKIL | | (LOCAL) | | PROBABILITY THAT DESIGNATOR OPERATOR IS KILLED BY DIRECT FIRE FROM TARGET VEHICLE. |
| PDOSUP | | (LOCAL) | | PROBABILITY THAT DESIGNATOR OPERATOR IS SUPPRESSED DUE TO OBSCURANTS CREATED BY NEAR MISSES CAUSED BY DIRECT FIRE FROM TARGET VEHICLE. |
| PDSTKL | | MISC | | PROBABILITY THAT DUST FROM HE ARTILLERY FIRE IS SUFFICIENT TO OBSCURE TARGET AND PREVENT USE OF COPPERHEAD. |
| PE | | (LOCAL) | | PROBABILITY THAT ROUND ENGAGES TARGET (I.E., SUFFICIENT REFLECTED LASER ENERGY IS PICKED UP BY THE SEEKER AND THE TARGET IS WITHIN THE MANEUVER FOOTPRINT OF THE COPPERHEAD ROUND). |
| PECHK | | (LOCAL) | | USED TO DECIPHER PEDATA RECORD NAME FOR CASE HEADING OUTPUT. |
| PECODE | | RECNAM | | RECORD NAME KEY FOR PROBABILITY OF ENGAGEMENT DATA RECORD ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| PEDATA(4260) | | (LOCAL) | | ARRAY TO WHICH THE ELEMENTS OF THE PROBABILITY OF ENGAGEMENT (PE) DATA BLOCK ARE EQUIVALENCED. |
| PETBL(60,10,7) | | HIT | | PETBL(I,J,K) IS THE PROBABILITY THAT THE COPPERHEAD ROUND ENGAGES THE TARGET GIVEN INDEX VALUE I, VISIBILITY RANGE LIMIT J, AND DESIGNATOR TARGET RANGE K. |
| PET1R1 | | (LOCAL) | | PROBABILITY OF ENGAGEMENT VALUE FOR TIME 1 AND RANGE 1 WHEN INTERPOLATING IN PROBABILITY OF ENGAGEMENT DATA WITH IMUTS=1 |
| PET1R2 | | (LOCAL) | | SAME AS PET1R1 BUT FOR TIME 1 AND RANGE 2. |
| PET2R1 | | (LOCAL) | | SAME AS PET1R1 BUT FOR TIME 2 AND RANGE 1. |
| PET2R2 | | (LOCAL) | | SAME AS PET1R1 BUT FOR TIME 2 AND RANGE 2. |
| PE1 | | (LOCAL) | | LOWER RANGE PROBABILITY OF ENGAGEMENT VALUE WHEN INTERPOLATING WITH RESPECT TO RANGE IN PROBABILITY OF ENGAGMENT DATA WITH IMUTS=2. |

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| PE2 | | (LOCAL) | | UPPER RANGE PROBABILITY OF ENGAGEMENT VALUE WHEN INTERPOLATING WITH RESPECT TO RANGE IN PROBABILITY OF ENGAGEMENT DATA WITH IMUTS=2. |
| PH | | (LOCAL) | | PROBABILITY ROUND HITS TARGET GIVEN THAT ROUND ENGAGES. |
| PK | | (LOCAL) | | PROBABILITY ROUND KILLS TARGET VEHICLE GIVEN THAT ROUND HITS TARGET VEHICLE. |
| PKTBL(10,2) | | HIT | | PKTBL(I,J) IS THE PROBABILITY OF KILL GIVEN A HIT FOR COPPERHEAD AGAINST TARGET TYPE I IN TARGET POSTURE J. |
| PR | | (LOCAL) | | USED TO BRACKET UNIFORM RANDOM NUMBER IN A WEATHER CLASS PROBABILITY TABLE. |
| PRBAEO | | (LOCAL) | | PROBABILITY OF ATTEMPTED ENGAGEMENT GIVEN AN OCCASION. |
| PRBKAE | | (LOCAL) | | PROBABILITY OF KILL GIVEN AN ATTEMPTED ENGAGEMENT. |
| PRBKO | | (LOCAL) | | PROBABILITY OF KILL GIVEN AN OCCASION. |
| PRBKS | | (LOCAL) | | PROBABILITY OF KILL GIVEN A SHOT. |
| PRBLOS(10) | | RANDOC | | PRBLOS(I) IS THE PROBABILITY THAT A SINGLE VEHICLE LOCATED AT RANDOM ALONG AN APPROACH PATH IN RANGE BRACKET I WILL BE WITH IN THE DESIGNATOR OPERATOR'S LINE-OF-SIGHT FOR AT LEAST THE CRITICAL TIME (TCRIT SECONDS). |
| PRBSAE | | (LOCAL) | | PROBABILITY OF SHOT GIVEN AN ATTEMPTED ENGAGEMENT. |
| PRBSO | | (LOCAL) | | PROBABILITY OF SHOT GIVEN AN OCCASION. |
| PRCFLS(2) | | WEATHR | | PRCFLS(I) IS THE PROBABILITY OF CLOUD FREE LINE-OF-SIGHT GIVEN CONDITION I. (I=1 INDICATES A CLOUD CEILING; I=2 INDICATES SCATTERED CLOUDS ONLY). |
| PRCLCG | | WEATHR | | PROBABILITY THAT THERE IS A CLOUD CEILING. |
| PRDOKC | | (LOCAL) | | PROBABILITY DESIGNATOR KILLED BY DIRECT FIRE GIVEN THAT DIRECT FIRE OCCURRED. |
| PRDOKL | | (LOCAL) | | PROBABILITY DESIGNATOR KILLED GIVEN AN OCCASION |
| PRDOW | | RSPTIM | | PROBABILITY THAT THE DESIGNATOR-OPERATOR IS WARNED TO BEGIN LASING IN TIME TO GUIDE THE FIRST ROUND OF THE FIRE MISSION TO THE TARGET. |

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| PRGCFL(11,2) | | WEATHR | | PRGCFL(I,J) IS THE PROBABILITY OF METEOROLOGICAL VISIBILITY RANGE LIMIT I GIVEN CLOUD FREE LINE-OF-SIGHT AND CONDITION J (J=1 INDICATES CLOUD CEILING; J=2 INDICATES SCATTERED CLOUDS) |
| PRHLOS(6) | | (LOCAL) | | PRHLOS(I) IS THE PROBABILITY THE LINE-OF-SIGHT EXISTS FROM DESIGNATOR TO TARGET FOR ROUND I. (I.E., LOS EXISTS BOTH BEFORE FIRING AND DURING TERMINAL PHASE OF TRAJECTORY). |
| PRIV | | (LOCAL) | | PROBABILITY THAT AT LEAST ONE TARGET IS IN VIEW FOR THE CRITICAL TIME (TCRIT SECONDS) DURING THE TERMINAL PHASE OF THE COPPERHEAD TRAJECTORY. |
| PRLOS | | (LOCAL) | | PROBABILITY OF LINE-OF-SIGHT TO A SINGLE TARGET FOR THE CRITICAL TIME DURATION (TCRIT SECONDS) DURING THE TERMINAL PHASE OF THE COPPERHEAD TRAJECTORY. |
| PRSDT | | RSPTIM | | PROBABILITY THAT DESIGNATOR TO FDC DIGITAL COMMUNICATION LINK IS OPERATING SUCCESSFULLY. |
| PRSPTM | | RSPTIM | SECONDS | PARAMETERIZED RESPONSE TIME. |
| PRSVT | | RSPTIM | | PROBABILITY THAT DESIGNATOR TO FDC VOICE COMMUNICATION IS SUCCESSFUL. |
| PRI | | (LOCAL) | | USED TO BRACKET RANDOM NUMBER IN A WEATHER CLASS PROBABILITY TABLE. |
| PSCODE | | RECNAM | | NAME RECORD KEY FOR TARGET POSTURE DATA RECORD ON WORD ADDRESSABLE RANDOM ACCESS MASS STORAGE FILE (TAPE 11). |
| PSMKKL | | SMOKED | | PROBABILITY THAT SMOKE ABORTS POTENTIAL COPPERHEAD FIRE MISSION. |
| PSTDAT(41) | | (LOCAL) | | ARRAY TO WHICH THE ELEMENTS OF THE TARGET POSTURE DISTRIBUTION DATA BLOCK ARE EQUIVALENCED. |
| PSTTBL(10,3) | | HIT | | PSTTBL(I,J) IS THE PROBABILITY THAT A MOVING TARGET AT RANGE I IS IN POSTURE J (WHERE J=1 INDICATES A COMPLETELY OBSCURED TARGET, J=2 INDICATES A FULLY EXPOSED TARGET, AND J=3 INDICATES A HULL DEFILADE TARGET). |
| PSTVAL(3) | | (LOCAL) | | PSTVAL(I) IS THE PROBABILITY THAT THE TARGET IS IN POSTURE I WHERE I=1,2 AND 3 RESPECTIVELY INDICATE THAT THE TARGET IS COMPLETELY OBSCURED, FULLY EXPOSED, AND HULL DEFILADE. |
| R(1) | | (LOCAL) | | GAMMA RANDOM DEVIATE. |

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| RECLBL(8) | | DISPLY | | RECLBL(I) IS AN ALPHANUMERIC CODE THAT FORMS PART OF THE RECORD NAME KEY FOR WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11) RECORDS OF TYPE I. (WHICH CORRESPOND TO KEYWORD 20+I). |
| REFL | | PEDESC (F.P.) | | REFLECTIVITY OF TARGET TO LASER ENERGY IN DESIGNATOR WAVE LENGTH. |
| RELIF | | MISC | | ROUND IN FLIGHT RELIABILITY. PROBABILITY THAT ROUND FUNCTIONS CORRECTLY. |
| RN | | (LOCAL) | | UNIFORMLY DISTRIBUTED RANDOM NUMBER. |
| RNDREC(6) | | SCCSS | | RNDREC(I) IS THE NUMBER OF TARGETS KILLED FOR THE CURRENT CASE BY THE ITH ROUND. (RNDREC(I) = NKILL(I).) |
| RNGCLB(11) | | RNGLOS | METERS | RNGCLB(I) IS THE UPPER BOUNDARY OF THE ITH RANGE CLASS FOR LINE-OF-SIGHT SEGMENT LENGTHS. |
| RNGK | | (LOCAL) | KM | RNGK IS THE CURRENT DESIGNATOR-TO-TARGET RANGE IN KILOMETERS |
| RNGNOW | | (LOCAL) | METERS | CURRENT DESIGNATOR-TO-TARGET RANGE IN METERS. |
| RNGPLS(10) | | RANDOC | METERS | RNGLOS(I) IS THE UPPER BOUNDARY OF THE ITH RANGE CLASS OF PRBLOS ARRAY. |
| RNGPST(10) | | HIT | METERS | RNGPST(I) IS THE DESIGNATOR-TO-TARGET RANGE FOR THE ITH POSTURE BREAKDOWN DISTRIBUTION IN THE PSTTBL ARRAY. |
| RNGTTF(20) | | HIT | METERS | RNGTTF(I) IS THE DESIGNATOR-TO-TARGET RANGE FOR THE ITH SET OF "TRUE TARGET FACTORS" IN THE TTF ARRAY. |
| RN1, RN2, . . RN22 | | (LOCAL) | | THE RANDOM NUMBERS RESULTING FROM CALLS OF URAN31 (I.E., RNi=URAN31 (IRi)) SOME OF THE RN'S (RN9 FOR EXAMPLE) ARE NO LONGER PRESENT IN THE PROGRAM). |
| ROCODE | | RECNAM | | RECORD NAME KEY FOR RANDOM OCCURRENCE DATA RECORD ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| RODATA(22) | | (LOCAL) | | ARRAY TO WHICH THE ELEMENTS OF THE RANDOM OCCURRENCE DATA BLOCK ARE EQUIVALENCED. |
| ROLBL(4,2) | | HEADNG | | ARRAY OF ALPHANUMERIC CONSTANTS USED TO CREATE LABELING OF LOS MODE SECTION OF OUTPUT. |
| RSCODE | | RECNAM | | RECORD NAME KEY FOR RESPONSE TIME DATA RECORD ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| RSPDAT(49) | | (LOCAL) | | ARRAY TO WHICH THE ELEMENTS OF THE RESPONSE TIME DATA BLOCK ARE EQUIVALENCED. |
| RSPLBL(3,3) | | HEADNG | | ARRAY OF ALPHANUMERIC CONSTANTS USED TO CREATE LABELING OF RESPONSE TIME SECTION OF OUTPUT. |
| RUMC | | RSPTIM | METERS | RANGE FROM TARGET UNMASK TO FOOTPRINT CENTROID. |
| SEGLNG | | (LOCAL) | METERS | LENGTH OF LINE-OF-SIGHT SEGMENT DRAWN FOR THE CURRENT REPLICATION. |
| SEGLOS(11,11) | | RNGLOS | NONE, METERS | SEGLOS(I,J) FOR J=1 IS THE ITH PROBABILITY VALUE USED FOR BRACKETING A UNIFORM RANDOM NUMBER WHEN INTERPOLATING IN CUMULATIVE LOS SEGMENT LENGTH DISTRIBUTION. SEGLOS(I,J) FOR J>1 IS THE ITH LINE-OF-SIGHT SEGMENT LENGTH VALUE USED FOR INTERPOLATING IN RANGE CLASS J-1. (SEGLOS(I,1) IS THE PROBABILITY THAT A LINE-OF-SIGHT SEGMENT IN RANGE CLASS J-1 IS LESS THAN OR EQUAL TO SEGLOS(I,J)). |
| SEP(5) | | SYMBOL | | AN ARRAY OF ALPHANUMERIC SYMBOLS THAT CAN BE USED AS KEYWORD OR OPTION WORD SEPARATORS (",","(",")", "/", "?") THEY ARE STORED IN A1 FORMAT. |
| SEQNML | LOGICAL | LOGFLG | | FLAG TO INDICATE WHETHER SEQUENCE NUMBERS (FROM CDC EDITOR) ARE PRESENT IN COLUMNS 73-78 OF INPUT FILE. (TRUE INDICATES SEQUENCE NUMBERS; FALSE INDICATES NO SEQUENCE NUMBERS). |
| SHRTEC | LOGICAL | LOGFLG | | FLAG TO INDICATE WHETHER SHORT OR LONG FORM OF SUBROUTINE ECHO'S OUTPUT IS TO BE USED. (SHRTEC=TRUE FOR SHORT FORM; SHRTEC=FALSE FOR LONG FORM). |
| SKSEN | | PEDESC F.P. | JOULES/$M^2$ | COPPERHEAD SEEKER SENSITIVITY. |
| SLCOM3 | LOGICAL | F.P. | | FLAG TO INDICATE WHETHER A COMMENT IS A PROGRAM GENERATED "TRIPLE $" COMMENT. (IF TRUE, IT IS; IF FALSE, IT IS NOT). |
| SLVERS | | STITLE | | VERSION NUMBER OF PROGRAM (INCREASED AS PROGRAM IS MODIFIED) |
| SMK2(3,3,2) | | SMOKED | TYPE 1 SMK ROUNDS | SMK2(I,J,K) IS THE NUMBER OF TYPE 1 SMOKE ROUNDS REQUIRED TO CREATE A SMOKE SCREEN FOR 30 MINUTES ACROSS 1 KILOMETER OF FRONT SUFFICIENT TO PREVENT THE USE OF COPPERHEAD AGAINST TARGETS IN THE SMOKED AREA GIVEN PASQUILL ATMOSPHERIC STABILITY CATEGORY 2*I OR 2*I-1, WIND SPEED J, AND RELATIVE HUMIDITY K. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| SMK5(3,3,2) | | SMOKED | TYPE 2 SMK ROUNDS | SAME AS SMK2 EXCEPT FOR TYPE 2 SMOKE ROUND. |
| SPCL(70) | LOGICAL | LOGFLG | | SPCL(I) IS TRUE IF A SPECIAL OPTION IS IN EFFECT FOR KEYWORD I; FALSE IF SPECIAL OPTION NOT IN EFFECT. |
| TBR | | TIME | SECONDS | TIME BETWEEN SUCCESSIVE ROUNDS FIRED ON A SINGLE REPLICATION. |
| TCRIT | | RANDOC | SECONDS | CRITICAL TIME. LENGTH OF TIME INTERVAL AT TERMINAL PHASE OF TRAJECTORY DURING WHICH COPPERHEAD SEEKER MUST BE TRACKING A SPOT DESIGNATED ON (OR NEAR) THE TARGET. |
| TF | | (LOCAL) | | FACTOR USED FOR INTERPOLATING WITH RESPECT TO RANGE IN PROBABILITY OF ENGAGEMENT TABLES WHEN IMUTS=2. |
| TFRAC | | (LOCAL) | | WEIGHTING FACTOR USED FOR INTERPOLATING WITH RESPECT TO RANGE IN TARGET POSTURE DISTRIBUTION TABLE. |
| TFRAC1 | | (LOCAL) | | WEIGHTING FACTOR USED FOR INTERPOLATING WITH RESPECT TO RANGE IN "TRUE TARGET FACTOR" TABLE. |
| TF1 | | (LOCAL) | | WEIGHT FACTOR APPLIED FOR SHORTER RANGE WHEN INTERPOLATING WITH RESPECT TO RANGE IN PROBABILITY OF ENGAGEMENT TABLES WITH IMUTS=1. |
| TF2 | | (LOCAL) | | WEIGHT FACTOR APPLIED FOR LONGER RANGE WHEN INTERPOLATING WHEN RESPECT TO RANGE IN PROBABILITY OF ENGAGEMENT TABLES WITH IMUTS=1. |
| TGTHD | | PEDESC (F.P.) | DEGREES | ANGLE BETWEEN FOOTPRINT CENTROID-TO-GUN LINE AND TARGET DIRECTION OF TRAVEL. |
| TIM | | RUNDAT | | ALPHANUMERIC DISPLAY OF TIME AT WHICH PROGRAM WAS RUN (OBTAINED FROM COMPUTER's INTERNAL CLOCK). |
| TIMNOW | | TIME | SECONDS | CURRENT TIME IN SECONDS SINCE TARGET UNMASKED FROM TERRAIN. |
| TIMRA | | TIME | SECONDS | TIME AT WHICH ROUND CURRENTLY BEING CONSIDERED WILL ARRIVE (IMPACT). |
| TIMRA1 | | TIME | SECONDS | TIME AT WHICH FIRST ROUND OF THIS REPLICATION WILL ARRIVE (IMPACT). |

| VARIABLE | TYPE | COMMONBLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| TMEAN(2,2) | | RSPTIM | SECONDS | TMEAN(I,J) IS THE MEAN TIME REQUIRED TO ENTER COPPERHEAD FIRE MISSION REQUEST IN DIGITAL MESSAGE DEVICE FOR CONDITION I, J. (I=1 INDICATES DAY, I=2 INDICATES NIGHT, J=1 INDICATES PRE-PLANNED TARGET, J=2 INDICATES TARGET OF OPPORTUNITY). |
| TMPLBL(8) | | DISPLY | | ALPHANUMERIC CONSTANTS USED TO LABEL OUTPUT HEADINGS WHEN TEMPORARY OR OVERRIDE OPTION IS USED. |
| TOF | | TIME | SECONDS | TIME OF FLIGHT FOR COPPERHEAD ROUND (TIME FROM FIRING UNTIL IMPACT). |
| TOFARY(16) | | FLTTIM | SECONDS | TOFARY(I) IS THE TIME OF FLIGHT OF COPPERHEAD ROUND (USING PREFERRED ELEVATION AND CHARGE) FOR GUN-TO-TARGET RANGE OF I KILOMETERS. |
| TR | | (LOCAL) | SECONDS | NOMINAL DESIGNATOR-FDC-BATTERY RESPONSE TIME. |
| TRARRY(3,2) | | RSPTIM | SECONDS | TRARRY(I,J) IS THE NOMINAL DESIGNATOR-FDC-BATTERY RESPONSE TIME FOR MISSION TYPE I AND COMMUNICATION TYPE J. (I=1 INDICATES PRE-PLANNED TARGET, I=2 INDICATES TARGET OF OPPORTUNITY, I=3 INDICATES PRIORITY PRE-PLANNED TARGET, J=1 INDICATES DIGITAL COMMUNICATION, J=2 INDICATES VOICE COMMUNICATION). |
| TRBAR | | RSPTIM | SECONDS | NOMINAL DESIGNATOR-FDC-BATTERY RESPONE TIME PLUS TIME OF FLIGHT. |
| TRBARI | | (LOCAL) | SECONDS | SAME AS TRBAR. |
| TSIGMA(2,2) | | RSPTIM | SECONDS | TSIGMA(I,J) IS THE STANDARD DEVIATION OF THE DISTRIBUTION OF TIME REQUIRED TO ENTER COPPERHEAD FIRE MISSION REQUEST IN DIGITAL MESSAGE DEVICE FOR CONDITION I,J. (I,J HAVE SAME MEANING AS FOR THE VARIABLE TMEAN). |
| TTF(3,20,2) | | HIT | | TTF(I,J,K) IS THE PROBABILITY COPPERHEAD HITS THE TARGET GIVEN SEEKER ENGAGEMENT WITH DESIGNATOR TYPE I, RANGE J, AND TARGET POSTURE K. (I=1,2,3 INDICATE DESIGNATOR TYPES 1,2,3 RESPECTIVELY; J INDICATES JTH RANGE OF RNGTTF ARRAY; K=1 INDICATES FULLY EXPOSED TARGET, K=2 INDICATES HULL DEFILADE TARGET). |
| TVEL | | TARGET | METERS/ SECONDS | TARGET SPEED. |

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| T1 | | (LOCAL) | SECONDS | FIRST TIME DELAY COMPONENT OF DELTAT. (ACCOUNTS FOR MULTIPLE VEHICLES AND MULTIPLE ROUNDS). ALSO, NOMINAL RESPONE TIME VALUE FOR PRINTING IN COMMENT. |
| T2 | | (LOCAL) | SECONDS | SECOND TIME DELAY COMPONENT OF DELTAT. (ACCOUNTS FOR ACTUAL RESPONE TIME AND DETECTION TIME VERSUS NOMINAL RESPONSE TIME). |
| T3 | | (LOCAL) | SECONDS, M/S | THIRD TIME DELAY COMPONENT OF DELTAT. (ACCOUNTS FOR POSITIONING OF FOOTPRINT WHEN RUMC IS PLAYED). ALSO, TARGET SPEED VALUE FOR PRINTING IN COMMENT. |
| T4 | | (LOCAL) | KM | GUN-TO-TARGET RANGE VALUE FOR PRINTING IN COMMENT. |
| T5 | | (LOCAL) | | TARGET REFLECTIVITY VALUE FOR PRINTING IN COMMENT. |
| T6 | | (LOCAL) | DEGREES | ANGLE T VALUE FOR PRINTING IN COMMENT. |
| T7 | | (LOCAL) | METERS | DEFLECTION BIAS (POINT OF CLOSEST APPROACH TO FOOTPRINT CENTROID) VALUE FOR PRINTING IN COMMENT. |
| T8 | | (LOCAL) | DEGREES | TARGET HEADING VALUE FOR PRINTING IN COMMENT. |
| T9 | | (LOCAL) | JOULES/$M^2$ | SEEKER SENSITIVITY VALUE FOR PRINTING IN COMMENT. |
| U(70) | | UVALUE | | U(I) IS A VALUE ASSOCIATED WITH KEYWORD I AND USED IN THE CURRENT CASE (EQUIVALENT TO UVALUE). |
| UDT | | UVALUE | | NO LONGER USED. |
| UIDIGT | | UVALUE | | UIDIGT=1.0 INDICATES DIGITAL DESIGNATOR-FDC COMMUNICATIONS; UIDIGT=2 INDICATES VOICE DESIGNATOR-FDC COMMUNICATIONS. |
| UIDN | | UVALUE | | UIDN=1 INDICATES DAY; UIDN=2 INDICATES NIGHT. |
| UMC | | UVALUE | | USED MISSION CODE. UMC=1,2,3 MEAN RESPECTIVELY PRE-PLANNED TARGET, TARGET OF OPPORTUNITY, PRIORITY PRE-PLANNED TARGET. |
| UPRT | | UVALUE | SECONDS | USED PARAMETERIZED RESPONSE TIME. |
| URAN31 | | FUNCTION VALUE | | URAN31 IS THE UNIFORM (PSEUDO-) RANDOM NUMBER GENERATOR USED IN THIS PROGRAM. |
| USERC | LOGICAL | F.P. | | USERC=TRUE INDICATES A USER'S COMMENT; USERC=FALSE INDICATES A PROGRAM GENERATED COMMENT. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| USMK(2) | | UVALUE | | USMK(1) IS THE NUMBER OF TYPE 1 SMOKE ROUNDS USED. USMK(2) IS THE NUMBER OF TYPE 2 SMOKE ROUNDS USED. |
| UVALUE(70) | | | | EQUIVALENT TO U ARRAY. |
| VEL | | F.P. | METERS/ SECONDS | TARGET SPEED. |
| VELTBL(2,2) | | RNGLOS | METERS/ SECONDS | VELTBL(I,J) IS TARGET SPEED IN CONDITIONS I,J. (I=1 FOR DAY, I=2 FOR NIGHT, J=1 FOR SUMMER, J=2 FOR WINTER). |
| VERDAT(2) | | STITLE | | DATE (ALPHANUMERIC CONSTANT) OF LAST CHANGE TO CURRENT PROGRAM. |
| VIS(11) | | WEATHR | KM | VIS(I) IS THE ITH METEOROLOGICAL VISIBILITY RANGE LIMIT VALUE USED IN THE WEATHER DATA. |
| VISLIM | | RANGE | KM | THE METEOROLOGICAL VISIBILITY RANGE LIMIT VALUE FOR THE CURRENT REPLICATION. |
| VSMK | | (LOCAL) | KM | LENGTH OF FRONT THAT CAN BE SMOKED WITH GIVEN NUMBER OF ROUNDS AND GIVEN WEATHER CONDITIONS. INTERMEDIATE VARIABLE IN CALCULATION OF PSMKKL. |
| W(6,11,2) | | WEATHR | | W(I,J,K) IS THE PROBABILITY (GIVEN A CLOUD CEILING) OF HAVING WEATHER WITH CLOUD CEILING ALTITUDE I, METEOROLOGICAL VISIBILITY RANGE LIMIT J, AND CLOUD FREE LOS CONDITION K (K=1 INDICATES CLOUD FREE LOS; K=2 INDICATES NO CLOUD FREE LOS). |
| WCFSUM(2) | | WEATHR | | SCFSUM(I) IS THE SUM OF ELEMENTS PRGCFL(1,I) THROUGH PRGCFL (NVL,I). (USED TO NORMALIZE RANDOM NUMBERS BEFORE SAMPLING FROM PRGCFL ARRAY). |
| WCODE | | RECNAM | | RECORD NAME KEY FOR WEATHER DATA RECORD ON WORD ADDRESSABLE MASS STORAGE FILE (TAPE 11). |
| WDATA(193) | | (LOCAL) | | ARRAY TO WHICH THE ELEMENTS OF THE WEATHER DATA BLOCK ARE EQUIVALENCED. |
| WK(6) | | LOCAL | | WORKING AREA ARRAY USED BY IMSL ROUTINE GGAMA. |
| WNDSPD(3,3) | | WEATHR | | WNDSPD(I,J) IS THE PROBABILITY OF HAVING JTH WINDSPEED GIVEN PASQUILL ATMOSPHERIC STABILITY CATEGORY 2*I OR 2*I-1. |

GLOSSARY - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| WSUM(2) | | WEATHR | | WSUM(I) IS THE SUM OF ELEMENTS W(1,1,I) THROUGH W(6,11,I). (USED TO NORMALIZE RANDOM NUMBERS BEFORE SAMPLING FROM W ARRAY). |
| X | | (LOCAL) | | USED AS INTERMEDIATE VARIABLE FOR CALLING FPRCNT. |
| XIN | | (F.P.) | | PERCENTAGE VALUE SENT TO FPRCNT FUNCTION FOR FORMATTING. |
| XMTTIM(2,2) | | RSPTIM | SECONDS | XMTTIM(I,J) IS THE TIME REQUIRED TO TRANSMIT THE REQUEST FOR COPPERHEAD FIRE FROM DESIGNATOR OPERATOR TO FDC USING COMMUNICATION MODE I FOR A MISSION TYPE J. (I=1 FOR DIGITAL COMMUNICATION, I=2 FOR VOICE; J=1 FOR PRE-PLANNED TARGET, J=2 FOR TARGET OF OPPORTUNITY). |
| XMVADF(2) | | RSPTIM | SECONDS | XMVADF(I) IS THE TIME REQUIRED TO TRANSMIT REQUEST FOR COPPERHEAD FIRE FROM DESIGNATOR OPERATOR TO FDC VIA VOICE AFTER TRYING DIGITAL (AND FAILING DIGITAL) FOR MISSION TYPE I. (I=1 FOR PRE-PLANNED TARGET, I=2 FOR TARGET OF OPPORTUNITY.) |
| XNV | | (LOCAL) | | NUMBER OF CURRENTLY UNKILLED VEHICLES IN TARGET. |
| XOUT | | (LOCAL) | | NAME OF VALUES OUTPUT AFTER SPECIAL FORMATTING. |
| XSMPL | | (F.P.) | | RANDOM SAMPLE VALUE RETURNED BY SUBROUTINE SMPLCD. |
| XVALUE(8,9,2) | | XVALUE | | XVALUE(I,J,1) IS THE ITH ALLOWABLE VALUE FOR THE JTH PARAMETER USED TO CONSTRUCT (OR DECIPHER) THE PEDATA RECORD NAME. XVALUE(I,J,2) IS THE CODE NUMBER CORRESPONDING TO XVALUE (I,J,1). (NOTE THAT NOT ALL POSSIBLE I,J COMBINATIONS NEED CORRESPOND TO VALUES ACTUALLY ALLOWED IN CURRENT CASES.) |
| XWTHR | | (LOCAL) | | ALPHANUMERIC ABBREVIATION FOR NAME OF MONTH WHOSE WEATHER DATA IS BEING USED. |
| Y | | (LOCAL) | | INTERMEDIATE VARIABLE NAME USED IN CALL TO FPRCNT. |
| YOUT | | (LOCAL) | | REFORMATTED PERCENTAGE VALUES. |
| TWTHR | | (LOCAL) | | ALPHANUMERIC TIME OF DAY VALUE FOR WEATHER DATA. |
| ZOUT | | (LOCAL) | | REFORMATTED PERCENTAGE VALUES. |

CHAPTER 13

# 13. CONVERSION TO OTHER COMPUTERS

The COPE model and its preprocessors were written for use on the US Army Ballistic Research Laboratories Control Data Corporation CYBER 76 (CDC 7600) computer. These programs use some of the features that are peculiar to CDC machines. This chapter points out some of the changes to COPE and its preprocessors that will be required to convert them to run on other machines.

## 13.1 Conversion To Other CDC Computers.

COPE and its preprocessors have been run on other CDC machines (a CDC 6600 at Aberdeen Proving Ground and a CDC 6400 and a CDC 6500 at FT Leavenworth, KS).

Two changes were required. The first was to change the Hollerith string delimiters from '(single quote) to whatever the host machine uses to delimit Hollerith strings (" double quote on the FT Leavenworth machine, ≠ on machines using CDC's standard, * at some other installations). This includes strings on STOP statements.

The second change was to define the cotangent function in PAM. The BRL CDC machines have the cotangent (COT) function as a library function; some other machines do not. This can be easily fixed by defining COT(X) = 1./TAN(X) as an in-line or statement function in PAM. Of course an alternate fix is to replace each occurrence of COT(X) by (1./TAN(X)).

Once these changes were made, the COPE program and its preprocessors ran without any trouble on the other CDC machines.

## 13.2 Conversion To Non-CDC Computers.

The author cannot anticipate every problem the user may encounter in converting COPE and its preprocessors to run on other machines; however, there are certain changes that will have to be made and of which the author is aware. The only machine (other than the CDC machines) that the author uses to any significant extent is the UNIVAC 1108, so most of the comments in this section deal with changes the author knows will have to be made to get the programs working on the UNIVAC machines.

13.2.1 Hollerith Delimiters and STOP's. The first change that may be required is to change the Hollerith string delimiters from ' to whatever is used on the host machine.

The CDC machines allow a STOP statement with a message (the message is in Hollerith string delimiters itself) which is printed when that STOP is encountered. Other machines may allow only a number with the stop, only a shorter message (CDC allows up to 70 characters; UNIVAC allows 6 characters), or nothing at all. So depending on the STOP

messages allowed, all of the STOP statements in COPE and its preprocessors may require changes.

13.2.2 A FORMATS and Alphanumeric Arrays. The CDC machines use a 60 bit word and a six bit character code so that each word holds 10 characters. Hence, most of the alphanumeric reads, writes, encodes, and decodes use A10 format and most data statements use 10H format to set alphanumeric values.

For machines that use shorter words or 8 bit character codes, the number of characters per word will probably be fewer. This means that the A10 formats in COPE and its preprocessors (and, indeed, all An formats where n> number of characters per word on the host machine) must be changed. A similar remark applies to all 10H (or nH with n as above) formats in data statements. Reducing these formats to the limits of the host machine will sometimes require that arrays containing the alphanumeric data be redimensioned. For example, the B array in subroutine INPUT is currently dimensioned B(8,10); if used on the UNIVAC 1108 which has 6 characters per word, it would have to be redimensioned B(14,10) so that each row could still contain a full card image (80 characters).

13.2.3 Word Addressable Mass Storage Files. The CDC system subroutines to open, read from, write to, and close a word addressable mass storage file are respectively OPENMS, READMS, WRITMS, and CLOSMS. The array INDEX11, the system subroutine SYSTEMC, and the user routines NOREC and NOREC2 are also connected with COPE's use of word addressable files.

The most convenient way to adapt COPE's handling of these files to the host machine is probably to write subroutines OPENMS, READMS, WRITMS, and CLOSMS that use the host machine's equivalent file type and commands, and which perform the same operations as on CDC.

In the specific case of the UNIVAC, for example, OPENMS would be a subroutine which first does a DEFINE FILE, then reads from the direct access file (using UNIVAC terminology -- this is equivalent to the CDC word addressable file) the current values of the INDX11 array. Subroutine READMS would search the INDX11 array for the desired record name and then when the name was found and the corresponding address v looked up, it would do a READ (11'v); if the record name were not found, NOREC would be called (or NOREC2 in PREPMS). Subroutine WRITMS would add a record name and address to INDX11 (or perhaps just modify the addresses if the record name were already present) and the do a WRITE (11'v). CLOSMS would do a WRITE (11'v) of INDX11 to TAPE11.

Non-CDC machines may also require redimensioning of INDX11 to allow record names to take up two (or more) words of characters. Also additional bookkeeping of record addresses may be required in PREPMS.

The calls to SYSTEMC and the defining of IRAY can be dispensed with if error prints are added that occur when attempting to read a non-existent record.

13.2.4 ENCODE and DECODE. Although most computers with FORTRAN IV compilers have ENCODE and DECODE statement's, there are some that may not and there are others that differ from CDC's.

If the host machine does not have ENCODE and DECODE statements, it would probably be worthwhile to write subroutines that perform the functions of ENCODE and DECODE and then replace all ENCODES and DECODES in COPE with calls to those subroutines.

If the host machine has ENCODE and DECODE a check should be made to see that they take the same arguments in the same order as the CDC ENCODE and DECODE statements and, if not, the necessary changes should be made.

13.2.5 DATE and TIME. CDC has two subroutines DATE and TIME that respectively give an 8 character date and a 9 character time. These are only called to obtain date and time values to be printed on the title page of the output. If the user's computer does not have equivalent subroutines, the calls to DATE and TIME may be deleted.

13.2.6 STRACE. The subroutine STRACE causes a printing of traceback information from the current subroutine back to the main program. This information gives the subroutine and line from which STRACE was called, the subroutine and line from which that subroutine was called, and so on back to the main program.

STRACE is available on both CDC and UNIVAC but it may not exist in some FORTRAN implementations (or it may have a different name). In either case, the user will need to check this point.

13.2.7 EOF. The CDC machine has a function called EOF. The value of EOF (IUNIT) is zero if the end of file has not been encountered on I/O unit IUNIT. After a READ, a test of EOF will yield a non-zero value if the end-of-file on that I/O unit has been encountered.

The CDC statements:

```
READ (IUNIT, 100) iolist
IF (EOF(IUNIT).NE.0.0) GO TO 99
```

are equivalent to:

```
READ (IUNIT, 100, END = 99) iolist
```

on most other machines.

13.2.8 COTANGENT. The PAM program uses the cotangent function (COT). If this is not a library function on the user's computer, then it should either be defined as a function (a statement function or external function) or else each occurrence of COT(X) should be replaced by (1./TAN(X)).

13.2.9 Im.n FORMAT. The CDC FORTRAN includes an Im.n format which specifies that an integer is to be written right justified in a field m characters wide with a minimum of n characters printed ($m \geq n$). This means that leading zeros are generated when the output value requires fewer than n digits. For example, if an integer variable with value 23 were to be written under I5.4 format, the result would be: b0023. Here b represents a blank so we have a field width of m=5 with n=4 characters printed even though 2 would have sufficed.

The Im.n is a standard format in FORTRAN 77, but since COPE is written in FORTRAN IV that may be of no help to the user whose FORTRAN IV compiler does not include it. On some machines a similar format (such as the UNIVAC Jn format) may be substituted; on other machines the user may have to simulate this format by using one of several different formats depending on the number of digits required to write the integer value.

## 13.3 GGAMA Function (IMSL).

The main COPE program uses the International Mathematical and Statistical Library (IMSL) function GGAMA. It is called from function GAMMA which is in turn called from GETTIM (it is used when the third method of playing response time is used).

If the IMSL is available on the user's computer, then it should be made available when COPE is loaded and run.

If the IMSL is not available to the user, then a function GGAMA should be written by the user or obtained from another library. It should take a random number seed IRN, an alpha parameter value (ALPHA), and return a gamma distributed random deviate R (with $\alpha$ = ALPHA and $\beta = 1$). The arguments IN and WK are arguments used by the IMSL routine and may or may not be used by the user's GGAMA.

CHAPTER 14

# 14. PREPMS PREPROCESSOR PROGRAM

## 14.1 General Remarks

The PREPMS preprocessor program performs three functions:

(1) to create or modify records on the word addressable data base file (TAPE 11) by reading input cards (or card images) and writing their contents (in appropriate format and order) to TAPE 11.

(2) to generate and print a list of the names of the current records on TAPE 11, and

(3) to generate and print a list of the PE data records currently on TAPE 11 along with their respective parameter values (see section 5.4 for the relation of 9 parameters to PE data record names).

Because the PREPMS program is quite straightforward in design and shares many subroutines with COPE, it is not documented as thoroughly as the main COPE program; however, this chapter gives enough information to set up the input, understand the output, and generally understand what the program does.

A re-reading of Chapter 5 may be helpful in understanding what follows.

## 14.2 PREPMS Program Design.

The PREPMS program consists of seven subprograms: the main program PREPMS, the block data subprogram BDATA2, the subroutines NOREC2, PENAME, and SEPRC2, and the functions IDCHAR and NUMRIC.

The subprograms PENAME, IDCHAR, and NUMRIC are identical to the subprograms of the same name in COPE. BDATA2 is merely a subset of BDATA1 of COPE; that is, it contains data statements that fill in the three common blocks LOGFLG, SYMBOL, and XVALUE as does BDATA1, but it does not fill in any of the other common blocks of BDATA1 as they are not needed by PREPMS.

The subroutine SEPRC2 is nearly identical to SEPREC of COPE. The only difference is that the message about sequence numbers in columns 73-78 of the input cards is written directly to OUTPUT instead of encoded and written via a call to COMMNT.

The subroutine NOREC2 is called (as a result of using CDC's SYSTEMC feature) whenever the program attempts to read a record named OPTNNUMS from TAPE 11 but does not find such a record. This should happen only when PREPMS is creating a new TAPE 11; when PREPMS is modifying an

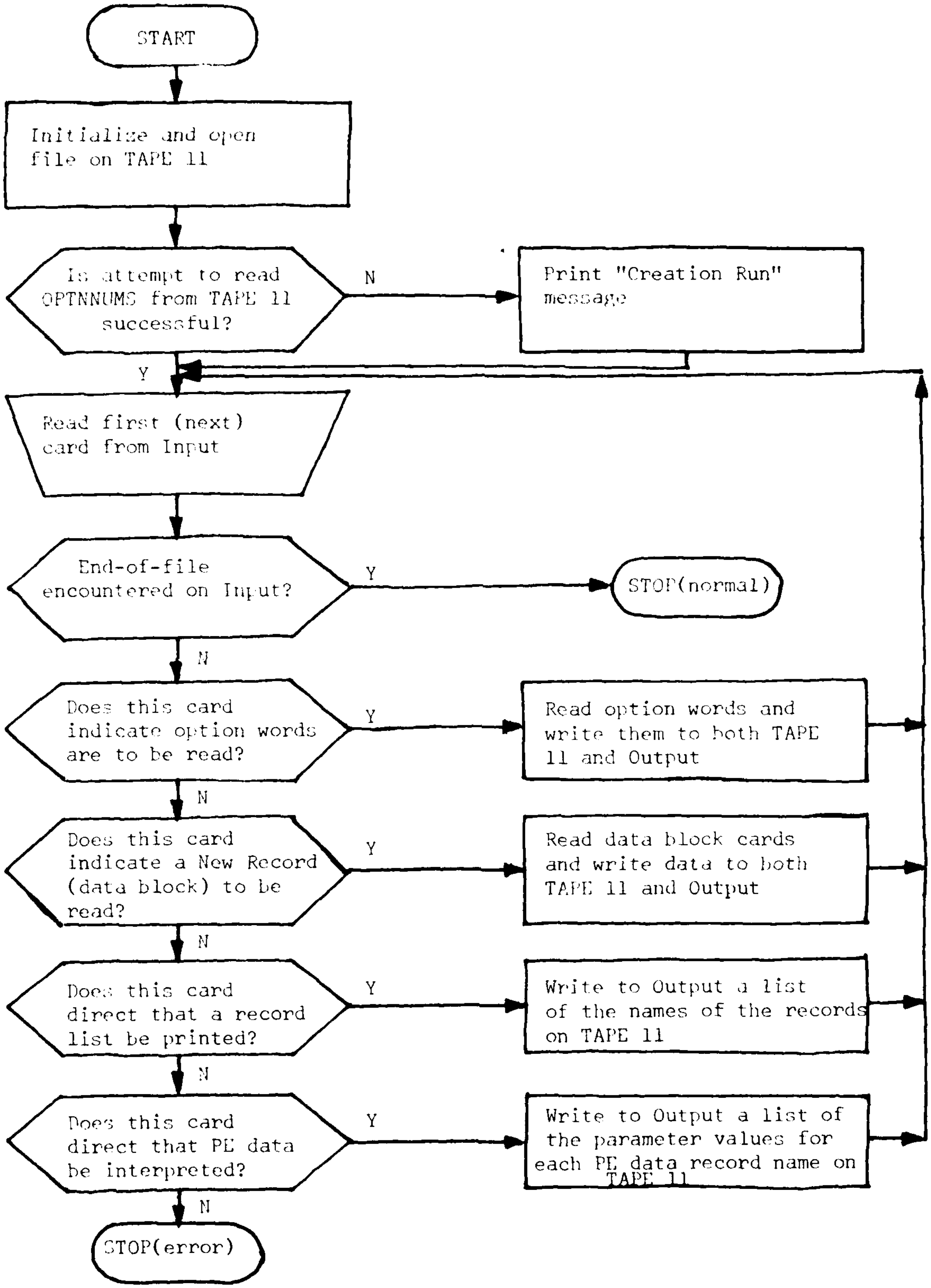



FIGURE 14-1 Logic Flow of PREPMS

existing TAPE11, there will already be an OPTNNUMS record on TAPE11. When no such record is found, NOREC2 is called; it then prints a message saying that the current job is a creation run and allows the program to continue.

The main program PREPMS is different from anything in COPE. Its main logic flow is shown in Figure 14-1. It begins by initializing (via DATA statements), calling SYSTEMC to set an error recovery procedure, and opening the file TAPE11. If it successfully reads the record named OPTNNUMS, it continues; if not, NOREC2 is called, a message is printed, and execution resumes. Next it attempts to read a card; if successful, the program continues; if unsuccessful, it has hit the end of file and a stop with normal termination message is issued.

The main program examines the card just read to determine what action to take next. Five possibilities exist:

(1) The card indicates a set of option words follows. In this case, the option words are read, printed on OUTPUT and written to TAPE 11.

(2) The card indicates a new record is to be read. In this case, the data for the new data block record is read, printed on OUTPUT, and written to TAPE11.

(3) The card indicates a list of the names of the records currently on TAPE11 is desired. In this case, such a list is generated and printed on OUTPUT.

(4) The card indicates PE data record names are to be interpreted. In this case, the names of the PE data records currently on TAPE11 are examined and the corresponding parameter values are determined. A list is then printed giving each PE data record's name and the corresponding parameter values.

(5) None of the above in which case a stop with error message occurs.

Note: The terms INPUT and OUTPUT refer respectively to the files TAPE5 and TAPE6 which correspond to I/o units 5 and 6. TAPE11 corresponds to I/o unit 11 and is another name for the word addressable data base file. Note that OUTPUT (or TAPE6) normally corresponds to a line printer and INPUT (or TAPE5) corresponds to either a punched card reader or a mass storage file of card images.

### 14.3 Glossary of PREPMS Variables.

The majority of the PREPMS variables have the same meaning as their namesakes in the COPE program. Therefore, the glossary in this section gives definitions only for those variables different from the main COPE model.

GLOSSARY OF PREPMS VARIABLES

| VARIABLES | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| CLABEL | REAL | (LOCAL) | | THE RECORD NAME OF THE OPTION WORD RECORD JUST READ. NORMALLY THIS WILL BE THE ABBREVIATED FORM OF THE KEYWORD TO WHICH THE OPTION WORDS BELONG. |
| ICASE | INTEGER | (LOCAL) | | THE TYPE NUMBER OF THE DATA BLOCK RECORD JUST READ. ICASE = 1 FOR WEATHER DATA, 2 FOR ACQUISITION DATA, 3 FOR RESPONSE TIME DATA, 4 FOR DIRECT FIRE SUPPRESSION DATA, 5 FOR RANDOM OCCURRENCE DATA, 6 FOR PEDATA, 7 FOR POSTURE DISTRIBUTION DATA, 8 FOR INVARIANT DATA, AND 9 FOR SADARM DATA (9 IS NOT USED IN RUNNING PREPMS FOR COPE). |
| ILIST(2000) | INTEGER | (LOCAL) | | ILIST (2I-1) IS THE NAME OF THE ITH RECORD ON TAPE 11 ILIST (2I) IS A NUMBER (PRINTED IN OCTAL) GIVING THE LENGTH AND ADDRESS ON DISK (PACKED) OF THE ITH RECORD ON TAPE 11. |
| IOUT | INTEGER | (LOCAL) | | IOUT IS A COUNTER USED TO RECORD THE NUMBER OF ILIST ENTRIES TO BE PRINTED. |
| IPECOD | INTEGER | (LOCAL) | | IPECOD IS THE NAME OF THE PE DATA RECORD CURRENTLY BEING EXAMINED TO DETERMINE PARAMETER VALUES. |
| IRLT1, IRLT2, ..., IRLT9 | INTEGER | (LOCAL) | WORDS | IRLT1 IS THE RECORD LENGTH OF TYPE 1 DATA BLOCK RECORDS (WEATHER DATA); IRLT2 IS THE RECORD LENGTH OF TYPE 2 DATA BLOCK RECORDS (ACQUISITION DATA); - - - -; IRLT9 IS THE RECORD LENGTH OF TYPE 9 DATA BLOCK RECORDS (SADARM DATA). |
| IUNIT | INTEGER | (LOCAL) | | I/O UNIT FROM WHICH PE DATA IS TO BE READ (MUST BE EITHER 4 OR 5). |
| I2 | INTEGER | (LOCAL) | | DESIGNATOR TYPE NUMBER (1 = GLLD, 2 = MULE, 3 = LTD). |
| NOPTNM(70) | INTEGER | (LOCAL) | | NOPTNM(I) IS THE NUMBER OF OPTION WORD CHARACTER STRINGS CURRENTLY AVAILABLE FOR USE WITH KEYWORD NUMBER I. |
| NOPT1 | INTEGER | (LOCAL) | | NUMBER OF THE KEYWORD CORRESPONDING TO THE SET OF OPTION WORDS JUST READ. |
| NREAD | INTEGER | (LOCAL) | | NUMBER OF OPTION WORD CHARACTER STRINGS TO BE READ IN THIS SET OF OPTION WORDS. |

GLOSSARY OF PREPMS VARIABLES - CONTINUED

| VARIABLES | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| OPT | REAL | (LOCAL) | | TEMPORARY NAME FOR THE VALUE TO BE LOADED INTO CHARA (1,IJ). IT IS THE NUMBER OF THE OPTION FOR ITS OPTION WORD POSITION (GIVEN BY CHARA (2,IJ) OR XLEV) TO WHICH THE CHARACTER STRING TO FOLLOW BELONGS. |
| PECHK | REAL | (LOCAL) | | PECHK IS THE FIRST FOUR CHARACTERS (LEFT JUSTIFIED WITH BLANK FILL) OF THE NAME OF THE RECORD ON TAPE11 CURRENTLY BEING EXAMINED TO DETERMINE WHETHER IT IS A PE DATA RECORD. |
| RLABEL | REAL | (LOCAL) | | NAME OF CURRENT DATA BLOCK RECORD. |
| T1, T3, T4, T5, T6, T7, T8, T9 | REAL | (LOCAL) | | THESE CORRESPOND RESPECTIVELY TO THE FIRST, THIRD, FOURTH, ----, NINTH ARGUMENTS OF SUBROUTINE PENAME (AND ITS ENTRY PEIDNT) T1 IS NOMINAL RESPONSE TIME, T3 IS TARGET VELOCITY, T4 IS GUN-TARGET RANGE, T5 IS TARGET REFLECTIVITY, T6 IS ANGLE T, T7 IS DEFLECTION BIAS, T8 IS TARGET HEADING, AND T9 IS SEEKER SENSITIVITY. |
| X | REAL | (LOCAL) | | TEMPORARY NAME FOR VALUE TO BE LOADED INTO NREAD (DEFINED ABOVE). |
| XLEV | REAL | (LOCAL) | | XLEV IS TEMPORARY NAME FOR VALUE TO BE LOADED INTO CHARA(2,IJ). IT IS THE OPTION WORD NUMBER FOR WHICH THE CHARACTER STRING TO FOLLOW IS A POSSIBLE CHOICE. |
| Y | REAL | (LOCAL) | | Y IS A TEMPORARY NAME FOR THE VALUE TO BE LOADED INTO NOPT1 (DEFINED ABOVE). |
| ZCASE | REAL | (LOCAL) | | TEMPORARY NAME FOR VALUE TO BE LOADED INTO ICASE (DEFINED ABOVE). |

To find the definition of a variable in PREPMS, one should first look in the glossary of this section. Then, if the variable is not found, look in the glossary of COPE variables in Chapter 12.

Note that all variables in common block SADARB are used only with the SADARM variant of COPE (called SOPE) and are not defined herein. This remark also applies to the WPNDAT array.

## 14.4 PREPMS INPUTS.

The PREPMS inputs are divided into four types:

(1) option word records
(2) data block records
(3) REPORT RECORD LIST directives, and
(4) INTERPRET PEDATA directives.

The second type (data block records) is further divided into eight types of data blocks. The sections concerning data block record types (sections 14.4.2(1) through 14.4.2(8)) also give instructions for setting up data to be read by COPE using the TEMPORARY feature (see section 4.5).

Note that if two or more option word records are read corresponding to the same keyword (see section 14.4.1) then only the last one read will have any effect. Similarly, if two or more data block records are read under the same record name, only the last one read will be retained on TAPE11. This means that any modifications to either type 1 or type 2 input (as listed above) require that the entire desired record contents be read (not just the additions).

Also, note that inputs of the four types may be arranged in any order provided that the integrity of the multiple card (or card image) input types (types 1 and 2) is preserved. However, types 3 and 4 only have effect on the file (TAPE11) as it exists at the time they are encountered; hence, they are usually put at the end of the input.

### 14.4.1 Option Word Records.

Each option word record consists of one card (or card image) of type 1 and NREAD cards of type 2 following the type 1 card.

The general form of the type 1 card is:

OPTION WORDS, keyword, m, n

or

OPTWDS, keyword, m, n

where: keyword is replaced by the character string giving the abbreviated form of the keyword to which the option words are to apply;

m = NREAD, a positive integer giving the number of type 2 cards to be read for this option word; and

n = NOPT1, a positive integer giving the keyword number (see section 5.3.9) of the keyword to which the option words are to apply.

The general form of each type 2 card is:

mm, nn, Option word character string

where: mm = NOPTWC, the number of the option word choice to which the character string belongs (NOPTWC is a positive integer);

nn = NOPTW, a positive integer equal to one plus the number of the option word for which the character string is an allowed choice; and

Option word character string is the actual character string allowed as a choice.

The subsequent use of option word character string in a COPE run as a choice for option word NOPTW-1 on a keyword input card for keyword number NOPT1 causes option number NOPTWC to be activated for option word NOPTW-1. (See example in section 5.3.8; for a second example, see appendixes C and D.)

Note: While blanks may be freely inserted in COPE inputs (keywords and option words) with no effect (the single exception being blanks occurring within 10 spaces after a "$" - see section 4.5.3), there should be no internal blanks in the character strings on type 2 cards.

Also, no more than 2 option word character strings applying to the same keyword are allowed to correspond to a given NOPTWC, NOPTW pair.

Finally, no more than 30 type 2 cards are allowed with any type 1 card.

14.4.2 Data Block Records. There are eight data block record types:

(1) Weather data
(2) Acquisition data
(3) Response time data
(4) Direct Fire Suppression data
(5) Random Occurrence data
(6) PE (probability of engagement data)
(7) Target Posture Distribution data
(8) Invariant data

The formatting of the input required for each of these types is presented in a separate subsection of this section.

Note that these eight data blocks correspond exactly to the eight data blocks that can be used with the TEMPORARY option in COPE. Furthermore, the formats used by COPE under the TEMPORARY option and by PREPMS are identical except for the first card (or card image) of each set. The sections that follow give the set-ups required for use with either program.

The first card (header card) in each case is free format: blanks may be freely inserted, fields are defined by separators (commas, for example). Subsequent cards of each data block type are formatted FORTRAN inputs and care must be taken to conform to the specified formats.

1. Weather Data Block Input.

When reading weather data into PREPMS for transfer onto TAPE11, the first card (which indicates a weather data record is to be input) must be of the form:

NEW RECORD, record name, 1

or

NEW REC, record name, 1

where record name is the name of the weather data record to be read. (See section 5.3.1 for the form of weather data record names).

When using the TEMPORARY option with COPE, the weather data is headed by a card of the form:

TEMPORARY, WEATHER or TEMPORARY, W

or

TEMP, WEATHER or TEMP, W

In either case (whether reading weather data in PREPMS or COPE), the remaining cards of the weather data are as follows:

WEATHER DATA CARD SET Card No. 2

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| NCC | | I5 | 1-5 | Number of different cloud ceiling altitude values used with the current weather data. (If NCC >6, then array dimension changes are required). |
| NVL | | I5 | 6-10 | Number of different visibility range limits used with the current weather data. (If NVL≠11, then array dimension changes and program modifications are usually required). |

WEATHER DATA CARD SET Card No. 3

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PRCLCG | | F10.4 | 1-10 | Probability that there is a cloud ceiling. |
| PRCFLS(1) | | F10.4 | 11-20 | Probability that there is a cloud free line-of-sight given that there is a cloud ceiling. |
| PRCFLS(2) | | F10.4 | 21-30 | Probability that there is a cloud free line-of-sight given that there is no cloud ceiling (i.e., scattered clouds only). |

WEATHER DATA CARD SET Card No. 4a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PRGCFL(1,1) | | F10.4 | 1-10 | PRGCFL(I,1) is the probability that the visibility range limit is equal to I kilometers given that there is a cloud ceiling and a cloud free line-of-sight. |
| PRGCFL(2,1) | | F10.4 | 11-21 | <u>Note</u>: The visibility range limit is considered to be I kilometers if (I-1/2)km < visibility range limit <(I+1/2)km. |
| ⋮ | | | | |
| PRGCFL(8,1) | | F10.4 | 71-80 | |

WEATHER DATA CARD SET Card No. 4b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PRGCFL(9,1) | | F10.4 | 1-10 | PRGCFL(I,1) is described on previous page. |
| ⋮ | | | | |
| PRGCFL(11,1) | | F10.4 | 21-30 | |

WEATHER DATA CARD SET Card No. 5a

| Variable name | UNITS | Format | columns | Description |
| --- | --- | --- | --- | --- |
| PRGCFL(1,2) | | F10.4 | 1-10 | PRGCFL(I,2) is the probability that the visibility range limit is equal to I kilometers given that there is no cloud ceiling and that there is a cloud free line-of-sight. The note on card 4a applies. |
| PRGCFL(2,2) | | F10.4 | 11-20 | |
| ⋮ | | | | |
| PRGCFL(8,2) | | F10.4 | 71-80 | |

WEATHER DATA CARD SET Card No. 5b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PRGCFL(9,2) | | F10.4 | 1-10 | PRGCFL(I,2) is described on previous page. |
| . | | | | |
| . | | | | |
| . | | | | |
| PRGCFL(11,2) | | F10.4 | 21-30 | |

WEATHER DATA CARD SET Card No. 6a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| W(1,1,1) | | F10.4 | 1-10 | W(I,J,1) is the probability of having the Ith cloud ceiling altitude and the Jth visibility range limit given that there is a cloud ceiling but no cloud free line-of-sight. |
| W(2,1,1) | | F10.4 | 11-20 | |
| W(3,1,1) | | F10.4 | 21-30 | |
| W(4,1,1) | | F10.4 | 31-40 | |
| W(5,1,1) | | F10.4 | 41-50 | |
| W(6,1,1) | | F10.4 | 51-60 | |

WEATHER DATA CARD SET Card No. 6b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| W(1,2,1) | | F10.4 | 1-10 | W(I,J,1) is as defined on previous page. |
| W(2,2,1) | | F10.4 | 11-20 | |
| W(3,2,1) | | F10.4 | 21-30 | |
| W(4,2,1) | | F10.4 | 31-40 | |
| W(5,2,1) | | F10.4 | 41-50 | |
| W(6,2,1) | | F10.4 | 51-60 | |

Cards 6c through 6k are similar to cards 6a and 6b except each successive card contains W(I,J,1) values for a higher value of J:

| | | | | | |
|---|---|---|---|---|---|
| card | 6a | contains | W(1,1,1) | through | W(6,1,1) |
| " | 6b | " | W(1,2,1) | " | W(6,2,1) |
| " | 6c | " | W(1,3,1) | " | W(6,3,1) |
| " | 6d | " | W(1,4,1) | " | W(6,4,1) |
| " | 6e | " | W(1,5,1) | " | W(6,5,1) |
| " | 6f | " | W(1,6,1) | " | W(6,6,1) |
| " | 6g | " | W(1,7,1) | " | W(6,7,1) |
| " | 6h | " | W(1,8,1) | " | W(6,8,1) |
| " | 6i | " | W(1,9,1) | " | W(6,9,1) |
| " | 6j | " | W(1,10,1) | " | W(6,10,1) |
| " | 6k | " | W(1,11,1) | " | W(6,11,1) |

Each of these cards contains six values,

$$\left\{ W(1,J,1),\ W(2,J,1),\ W(3,J,1),\ W(4,J,1),\ W(5,J,1),\ \text{and}\ W(6,J,1) \right\},$$

in 6F10.4 format (exactly as illustrated for cards 6a and 6b on the previous two pages).

WEATHER DATA CARD SET Card No. 7a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| W(1,1,2) | | F10.4 | 1-10 | W(I,J,2) is the probability of having the Ith cloud ceiling altitude and the Jth visibility range limit given that there is no cloud ceiling and no cloud free line-of-sight. |
| W(2,1,2) | | F10.4 | 11-20 | |
| W(3,1,2) | | F10.4 | 21-30 | |
| W(4,1,2) | | F10.4 | 31-40 | |
| W(5,1,2) | | F10.4 | 41-50 | |
| W(6,1,2) | | F10.4 | 51-60 | |

WEATHER DATA CARD SET Card No. 7b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| W(1,2,2) | | F10.4 | 1-10 | W(I,J,2) is as defined on previous page. |
| W(2,2,2) | | F10.4 | 11-20 | |
| W(3,2,2) | | F10.4 | 21-30 | |
| W(4,2,2) | | F10.4 | 31-40 | |
| W(5,2,2) | | F10.4 | 41-50 | |
| W(6,2,2) | | F10.4 | 51-60 | |

WEATHER DATA CARD SET Card No. 7c - 7k

Cards 7c through 7k are similar to cards 7a and 7b except each successive card contains W(I,J,2) values for a higher value of J:

| | | | | | |
|---|---|---|---|---|---|
| card | 7a | contains | W(1,1,2) | through | W(6,1,2) |
| " | 7b | " | W(1,2,2) | " | W(6,2,2) |
| " | 7c | " | W(1,3,2) | " | W(6,3,2) |
| " | 7d | " | W(1,4,2) | " | W(6,4,2) |
| " | 7e | " | W(1,5,2) | " | W(6,5,2) |
| " | 7f | " | W(1,6,2) | " | W(6,6,2) |
| " | 7g | " | W(1,7,2) | " | W(6,7,2) |
| " | 7h | " | W(1,8,2) | " | W(6,8,2) |
| " | 7i | " | W(1,9,2) | " | W(6,9,2) |
| " | 7j | " | W(1,10,2) | " | W(6,10,2) |
| " | 7k | " | W(1,11,2) | " | W(6,11,2) |

Each of these cards contains six values,

$\left\{ \text{W(1,J,2), W(2,J,2), W(3,J,2), W(4,J,2), W(5,J,2), and W(6,J,2)} \right\}$,

in 6F10.4 format (exactly as illustrated for cards 7a and 7b on the previous two pages).

WEATHER DATA CARD SET          Card No. 8a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| VISLIM(1) | km. | F10.4 | 1-10 | VISLIM (I) is the Ith visibility range limit value. |
| VISLIM (2) | km. | F10.4 | 11-20 | Any choice other than VISLIM(I) = I kilometers will require program |
| VISLIM (3) | km. | F10.4 | 21-30 | changes to COPE. |
| . | . | . | | |
| . | . | . | | |
| . | . | . | | |
| . | . | . | | |
| VISLIM (8) | km. | F10.4 | 71-80 | |

WEATHER DATA CARD SET Card No. 8b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| VISLIM (9) | km. | F10.4 | 1-10 | VISLIM (I) is as described on previous page. |
| . | . | . | | |
| . | . | . | | |
| . | . | . | | |
| VISLIM (11) | km. | F10.4 | 21-30 | |

WEATHER DATA CARD SET Card No. 9

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| CLOUD (1) | meters | F10.4 | 1-10 | CLOUD (I) is the Ith cloud ceiling altitude in meters used for the current set of weather data. |
| CLOUD (2) | meters | F10.4 | 11-20 | |
| CLOUD (3) | meters | F10.4 | 21-30 | |
| CLOUD (4) | meters | F10.4 | 31-40 | |
| CLOUD (5) | meters | F10.4 | 41-50 | |
| CLOUD (6) | meters | F10.4 | 51-60 | |

WEATHER DATA CARD SET Card No. 10

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PASQL (1) | | F10.4 | 1-10 | PASQL(I) is the probability of occurrence of the Ith Pasquill atmospheric stability category. (Category A corresponds to I=1, B to I=2, etc.) |
| PASQL (2) | | F10.4 | 11-20 | |
| PASQL (3) | | F10.4 | 21-30 | |
| PASQL (4) | | F10.4 | 31-40 | |
| PASQL (5) | | F10.4 | 41-50 | |
| PASQL (6) | | F10.4 | 51-60 | |

WEATHER DATA CARD SET          Card No. 11

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| WNDSPD (1) | | F10.4 | 1-10 | WNDSPD (I) is the probability that the windspeed is in speed bracket I. (Speed bracket I=1 includes windspeeds from 0 to 5 knots, I=2 includes from 5 to 15 knots and I=3 includes 15 knots and over). |
| WNDSPD (2) | | F10.4 | 11-20 | |
| WNDSPD (3) | | F10.4 | 21-30 | |

WEATHER DATA CARD SET Card No. 12

| Variable name | UNITS | Format | columns | Description |
| --- | --- | --- | --- | --- |
| HUMID (1) | | F10.4 | 1-10 | HUMID (1) is the probability that the relative humidity is less than 65%; HUMID (2) is the probability that the relative humidity is greater than (or equal to) 65%. |
| HUMID (2) | | F10.4 | 11-20 | |

2. Acquisition Data Block Input.

When reading acquisition data into PREPMS for transfer onto TAPE11, the first card (which indicates an acquisition data record is to be read) must be of the form:

NEW RECORD, record name, 2

or

NEW REC, record name, 2

where record name is the name of the acquisition data record to be read (see section 5.3.2 for the form of acquisition data record names).

When using the TEMPORARY option with COPE, the acquisition data is headed by a card of the form:

TEMPORARY, ACQUISITION RANGE DISTRIBUTION

or

TEMPORARY, ACQRNGDIST

or

TEMP, ACQUISITION RANGE DISTRIBUTION

or

TEMP, ACQRNGDIST

In either case (whether reading acquisition data in PREPMS or COPE), the remaining cards of the acquisition data are as follows:

ACQUISITION DATA CARD SET Card No. 2

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| NRP | | I5 | 1-5 | Number of range points used in the CRNGD array (Cumulative distribution of acquisition (unmasking) ranges.) |
| NPP | | I5 | 5-10 | Number of points used within each range class to describe cumulative distribution of line-of-sight segment lengths. |
| NRNGCL | | I5 | 11-15 | Number of range class boundary value to be used (equals one less than the number of range classes) Note: Zero is always assumed to be the lower boundary of the shortest range class so it need not be explicitly entered as input nor counted as a boundary value for NRNGCL. |

ACQUISITION DATA CARD SET Card No. 3a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| CRNGD(1,1) | | F10.4 | 1-10 | CRNGD(I,1) is the probability that the designator-to-target range at time of target unmasking is less than or equal to the value in CRNGD(I,2) |
| CRNGD(2,1) | | F10.4 | 11-20 | |
| CRNGD(3,1) | | F10.4 | 21-30 | Note: CRNGD(1,1) = 0.0 and CRNGD(NRP,1) = 1.0 must be satisfied. Also, the entries must increase as I increases. |
| / | | / | / | |
| CRNGD(8,1) | | F10.4 | 71-80 | |
| | | | | Note: If NRP $\leq$ 8, only one card of this type will be needed; if NRP > 8, then a second card must continue with CRNGD(9,1) through CRNGD(NRP,1). NRP cannot exceed 11 with current dimensions. |

ACQUISITION DATA CARD SET Card No. 3b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| CRNGD(1,2) | meters | F10.4 | 1-10 | CRNGD(I,2) is the range value corresponding to |
| CRNGD(2,2) | meters | F10.4 | 11-21 | the cumulative probability value in CRNGD(I,1). The values must increase as I increases. |
| ⋮ | ⋮ | ⋮ | ⋮ | |
| CRNGD(8,2) | meters | F10.4 | 71-80 | Note: If NRP ≤ 8, only one card of this type will be needed; if NRP > 8, then a second card must continue with CRNG( 9,2) through CRNGD(NRP,2) NRP cannot exceed 11 with current dimensions. |

ACQUISITION DATA CARD SET    Card No. 4a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| SEGLOS(1,1) | | F10.4 | 1-10 | SEGLOS(I,1) is the probability that the length of a line-of-sight segment in range class K is less than or equal to SEGLOS(I, K+1) meters. |
| SEGLOS(2,1) | | F10.4 | 11-20 | **Note:** SEGLOS(1,1) = 0.0; SEGLOS(NPP,1) = 1.0 and SEGLOS(I,1)< SEGLOS(I+1,1) for I = 1,2, --- NPP-1. |
| SEGLOS(8,1) | | F10.4 | 71-80 | If NPP ≤8, only one card of this type will be needed; if NPP >8, then a second card must continue with SEGLOS (9,1) through SEGLOS(NPP,1). NPP cannot exceed 11 with current dimensions. |

ACQUISITION DATA CARD SET Card No. 4b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| SEGLOS(1,2) | meters | F10.4 | 1-10 | SEGLOS(I,2) is the line-of-sight segment length value corresponding to the cumulative probability value in SEGLOS(I,1) when the range from D.O. to target unmask falls in the first range class. |
| SEGLOS(2,2) | meters | F10.4 | 11-20 | |
| SEGLOS(2,3) | meters | F10.4 | 21-30 | |
| / | | / | / | Note: SEGLOS(I,2)< SEGLOS(I+1,2) for I = 1,2,---, NPP-1. |

Note:

If NPP <8, only one card of this type is needed; if NPP >8, then a second card must continue with SEGLOS(9,2) through SEGLOS(NPP,2). NPP cannot exceed 11 with current dimensions.

ACQUISITION DATA CARD SET Card No. 4c - 4k

The SEGLOS array is read from additional cards similar to 4b up to a maximum of 11 cards (or card pairs if NNP >8).

Each card (or card pair if NPP >8) has the following content:

| | | | | |
|---|---|---|---|---|
| card 4a | contains | SEGLOS(1,1) | through | SEGLOS(NPP,1) |
| " 4b | " | SEGLOS(1,2) | " | SEGLOS(NPP,2) |
| " 4c | " | SEGLOS(1,3) | " | SEGLOS(NPP,3) |
| " 4d | " | SEGLOS(1,4) | " | SEGLOS(NPP,4) |
| " 4e | " | SEGLOS(1,5) | " | SEGLOS(NPP,5) |
| " 4f | " | SEGLOS(1,6) | " | SEGLOS(NPP,6) |
| " 4g | " | SEGLOS(1,7) | " | SEGLOS(NPP,7) |
| " 4h | " | SEGLOS(1,8) | " | SEGLOS(NPP,8) |
| " 4i | " | SEGLOS(1,9) | " | SEGLOS(NPP,9) |
| " 4j | " | SEGLOS(1,10) | " | SEGLOS(NPP,10) |
| " 4k | " | SEGLOS(1,11) | " | SEGLOS(NPP,11) |

If NPP >8 then each card above is replaced by a card pair. In this case, the first card of each pair contains eight values and the second card contains NPP-8 values (each card is in 8F10.4 format). SEGLOS(1,J) through SEGLOS(NPP,J) contain (for J$\geq$2) the line-of-sight segment lengths in meters for range class J-1 corresponding respectively to the cumulative probability values in SEGLOS(1,1) through SEGLOS(NPP,1). The total number of cards (or card pairs if NPP>8) will be NRNGCL + 1 (one for probabilities plus one for each range class). Hence, if NRNGCL = 10, all cards 3a through 3k will be used; but, if NRNGCL <10, then only the first NRNGCL + 1 cards (or card pairs if NPP >8) shown above will be used.

ACQUISITION DATA CARD SET Card No. 5

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| RNGCLB(1) | meters | F10.4 | 1-10 | RNGCLB(I) is the Ith non-zero range class boundary. That is, range class 1 runs from zero to RNGCLB(1) meters and, for I > 1, range class I runs from RNGCLB(I-1) meters to RNGCLB(I) meters.<br>(When the DO-to-target unmask range lies in range class I, then the (I + 1)st row of the SEGLOS array is sampled to obtain LOS segment lengths.)<br>Note that<br>$0 < RNGCLB(1) < \ldots < RNGCLB(I) \ldots < RNGCLB(I+1) < \ldots < \infty$.<br>Also note that if NRNGCL $\leq 8$, then only one card of this type is needed; but, if NRNGCL> 8, then a second card must continue with RNGCLB(9) through RNGCLB(NRNGCL). Finally, RNGCLB cannot exceed 10 with current dimensioning. |
| RNGCLB(2) | meters | F10.4 | 11-20 | |
| ⋮ | ⋮ | ⋮ | ⋮ | |
| RNGCLB(8) | meters | F10.4 | 71-80 | |

ACQUISITION DATA CARD SET    Card No. 6a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| VELTBL(1,1) | M/S | F10.4 | 1-10 | VELTBL(I,J) gives the target velocity default value in the current terrain under |
| VELTBL(2,1) | M/S | F10.4 | 11-20 | day/night condition I and weather condition J.<br>I = 1 for day time<br>2 for nighttime<br><br>J = 1 for "good" weather<br>2 for "bad" weather |

ACQUISITION DATA CARD SET    Card No. 6b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| VELTBL(1,2) | M/S | F10.4 | 1-10 | (see previous page) |
| VELTBL(2,2) | M/S | F10.4 | 11-20 | |

3. Response Time Data Block Input.

When reading response time data into PREPMS for transfer to TAPE11, the first card (which indicates that a response time data record is to be read) must be of the form:

NEW RECORD, record name, 3

or

NEW REC, record name, 3

where record name is the name of the response time record to be read (see section 5.3.3. for response time record names).

When using the TEMPORARY option with COPE, the response time data is headed by a card of the form:

TEMPORARY, RESPONSE TIME

or

TEMPORARY, RESPTIME

or

TEMP, RESPONSE TIME

or

TEMP, RESPTIME

Note: When the first method (section 2.2.9) of modeling delay times in COPPERHEAD mission communication and processing is played, in COPE, the response time distribution used is the one defined by data statements in BDATA1 (namely, the FSRT array - see section 4.7.6).

Because this array is sampled to obtain response times, the values on cards 4, 5, and 6 of this card set are not used. However, cards must still be input (though they may be blank) so that the statements which read the later cards of this set will pick up the correct data.

The record name used with the data for the first method of playing response time must be RSPDAT0123 unless one makes code changes in COPE to allow for others.

In any case (whether reading response time data in PREPMS or COPE), the remaining cards of the response time data are as follows:

RESPONSE TIME CARD SET Card No. 2

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| NDT | | I5 | 1-5 | Number of points to be entered in each row of DETTMA array. NDT ≤10 with current dimensions. |

RESPONSE TIME CARD SET Card No. 3a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| DETTMA (1,1) | | F10.4 | 1-10 | DETTMA (I,1) is the probability that the "detection" time is less than or equal to DETTMA(I,2) seconds. |
| DETTMA (2,1) | | F10.4 | 11-20 | |
| ⋮ | | ⋮ | | Note: DETTMA (1,1) = 0.0 and DETTMA (I) < DETTMA (I +1) |
| DETTMA(8,1) | | F10.4 | 71-80 | Also note that if NDT < 8, only one card of this type is required; if NDT > 8, then a second card is required with DETTMA (9,1) through DETTMA (NDT,1). |

RESPONSE TIME CARD SET Card No. 3b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| DETTMA (2,1) | seconds | F10.4 | 1-10 | DETTMA (I,2) is the "detection" time in seconds corresponding to DETTMA (I,1). |
| DETTMA (2,2) | seconds | F10.4 | 11-20 | Note: DETTMA (I) < DETTMA (I+1) |
| ⋮ | ⋮ | ⋮ | ⋮ | Also, if NDT ≤ 8 only one card is needed; if NDT > 8, then DETTMA (9,2) through DETTMA (NDT,2) continue on a second card. |
| DETTMA (8,2) | seconds | F10.4 | 71-80 | |

RESPONSE TIME CARD SET Card No. 4

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| BATRTM (1) | seconds | F10.4 | 1-10 | BATRTM (I) is the response time at the battery for a Copperhead mission of type I. |
| BATRTM (2) | seconds | F10.4 | 11-20 | I = 1 for preplanned targets |
| BATRTM (3) | seconds | F10.4 | 21-30 | I = 2 for targets of opportunity<br>I = 3 for priority preplanned targets |
| XMVADF (1) | seconds | F10.4 | 31-40 | XMVADF (I) is the time required to transmit message via voice commo after failing to transmit via digital commo. |
| XMVADF (2) | seconds | F10.4 | 41-50 | I = 1 for pre-planned targets (priority or not)<br>I = 2 for targets of opportunity |
| | | | | *Note: The first way of modeling delay* times does not use the data on cards 4, 5, and 6; hence, they may be left blank.<br>The second way of modeling delay times does not use any RESPONSE TIME data block values, so no record is prepared corresponding to the PARAM option.<br>Finally, the third delay time model uses all data in the data block.<br>See sections 2.2.9, 4.4.15 and 9.23 for further details on the three delay time models. |

RESPONSE TIME CARD SET Card No. 5

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| BCSPTM (1,1) | seconds | F10.4 | 1-10 | BCSPTM (I,J) is the battery computer system processing time for commo type I and mission type J. |
| BCSPTM (2,1) | seconds | F10.4 | 11-20 | |
| BCSPTM (1,2) | seconds | F10.4 | 21-30 | I = 1 for digital commo<br>I = 2 for voice commo |
| BCSPTM (2,2) | seconds | F10.4 | 31-40 | J = 1 for pre-planned target (priority or not)<br>J = 2 for target of opportunity |
| XMTTIM (1,1) | seconds | F10.4 | 41-50 | XMTTIM (I,J) is the commo transmission time for commo type I and and mission type J where I and J are as for BCSPTM above. |
| XMTTIM (2,1) | " | F10.4 | 51-60 | |
| XMTTIM (1,2) | " | F10.4 | 61-70 | |
| XMTTIM (2,2) | " | F10.4 | 71-80 | |

RESPONSE TIME CARD SET Card No. 6

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| TMEAN (1,1) | seconds | F10.4 | 1-10 | TMEAN (I,J) is the mean time for DO to enter data in the digital message device in day/night condition I and mission type J. |
| TMEAN (2,1) | seconds | F10.4 | 11-20 | I = 1 for day time |
| TMEAN (1,2) | seconds | F10.4 | 21-30 | I = 2 for night |
| TMEAN (2,2) | seconds | F10.4 | 31-40 | J = 1 for pre-planned target (priority or not) J = 2 for target of opportunity |
| TSIGMA (1,1) | seconds | F10.4 | 41-50 | TSIGMA (I,J) is the standard deviation of the distribution of times required for the DO to enter data in digital message device under day/night condition I and mission type J. |
| TSIGMA (2,1) | seconds | F10.4 | 51-60 | I and J are as defined above for TMEAN. |
| TSIGMA (1,2) | seconds | F10.4 | 61-70 | |
| TSIGMA (2,2) | seconds | F10.4 | 71-80 | |

RESPONSE TIME CARD SET Card No. 7

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| TRARRY (1,1) | seconds | F10.4 | 1-10 | TRARRY (I,J) is the nominal response time for mission type I |
| TRARRY (2,1) | seconds | F10.4 | 11-20 | and commo type J. |
| TRARRY (3,1) | seconds | F10.4 | 21-30 | I = 1 for pre-planned targets<br>I = 2 for targets of opportunity<br>I = 3 for priority pre-planned |
| TRARRY (1,2) | seconds | F10.4 | 31-40 | targets. |
| TRARRY (2,2) | seconds | F10.4 | 41-50 | J = 1 for digital commo |
| TRARRY (3,2) | seconds | F10.4 | 51-60 | J = 2 for voice commo |
| | | | | Note: These nominal response times are <u>not</u> *to include time-of-flight.* (TOF is <u>added</u> in COPE itself to obtain TRBAR). |

4. <u>Direct Fire Suppression Data Block Input</u>.

When reading direct fire suppression data into PREPMS for transfer to TAPE 11, the first card (which indicates that a direct fire suppression data record is to be read) must be of the form:

<u>NEW RECORD, record name, 4</u>

or

<u>NEW REC, record name, 4</u>

where record name is the name of the direct fire suppression record to be read (see section 5.3.4 for direct fire suppression record names).

When using the <u>TEMPORARY</u> option with COPE, the direct fire suppression data is headed by a card of the form:

<u>TEMPORARY DIRECT FIRE SUPPRESSION</u>

or

<u>TEMPORARY, DFIRE SUPPR</u>

or

<u>TEMP, DIRECT FIRE SUPPRESSION</u>

or

<u>TEMP, DFIRE SUPPR</u>

In either case (whether reading direct fire suppression data in PREPMS or COPE), the remaining cards of the direct fire suppression data are as follow:

DIRECT FIRE SUPPRESSION CARD SET Card No. 2

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| NDFSP | | I5 | 1-5 | Number of points to be entered for each of the three rows of the DFDOKL array. (Number of range points for DFDOKL array).<br><br>With current dimensions, NDFSP $\leqslant$ 10. |

DIRECT FIRE SUPPRESSION CARD SET Card No. 3a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| DFDOKL (1,1) | meters | F10.4 | 1-10 | DFDOKL (I,1) is the range in meters corresponding to a probability of DO being suppressed (obscured) of DFDOKL (I,2) and a probability of DO being killed of DFDOKL (I,3) |
| DFDOKL (2,1) | meters | F10.4 | 11-20 | DFDOKL (1,1) = 0.0 and DFDOKL (I,1) $<$ DFDOKL (I+1,1). |
| ⋮ | ⋮ | ⋮ | ⋮ | |
| DFDOKL (8,1) | meters | F10.4 | 71-80 | If NDFSP $>$ 8, then a second card is required for DFDOKL (9,1) through DFDOKL (NDFSP,1). |

DIRECT FIRE SUPPRESSION CARD SET Card No. 3b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| DFDOKL (1,2) | | F10.4 | 1-10 | DFDOKL (I,2) is the probability that direct fire dust from the target at range DFDOKL (I,1) obscures the DO (but he is not killed). |
| DFDOKL (2,2) | | F10.4 | 11-20 | |
| DFDOKL (8,2) | | F10.4 | 71-80 | If NDFSP > 8, then a second card is needed for DFDOKL (9,2) through DFDOKL (NDFSP,2). |

DIRECT FIRE SUPPRESSION CARD SET Card No. 3c

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| DFDOKL (1,3) | | F10.4 | 1-10 | DFDOKL (I,3) is the probability that direct fire from the target at range DFDOKL (I,1) kills the DO. |
| DFDOKL (2,3) | | F10.4 | 11-20 | |
| ⋮ | | ⋮ | ⋮ | If NDFSP > 8, then a second card is needed for DFDOKL (9,3) through DFDOKL (NDFSP,3) |
| DFDOKL (8,3) | | F10.4 | 71-80 | |

5. Random Occurrence Data Block Input.

When reading random occurrence data in PREPMS for transfer to TAPE 11, the first card (which indicates that a random occurrence data record is to be read) must be of the form:

NEW RECORD, record name, 5

or

NEW REC, record name, 5

where record name is the name of the random occurrence record to be read (see section 5.3.5 for random occurrence record names).

When using the TEMPORARY option with COPE, the random occurrence data is headed by a card of the form:

TEMPORARY,RANDOM OCCURRENCE DISTRIBUTION

or

TEMPORARY, RODIST

or

TEMP, RANDOM OCCURRENCE DISTRIBUTION

or

TEMP, RODIST

In either case (whether reading random occurrence data in PREPMS or COPE), the remaining cards of the random occurrence data are as follow:

RANDOM OCCURRENCE DATA CARD SET Card No. 2

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| NPLOSR | | I5 | 1-5 | Number of range classes for probability of LOS data for random occurrence.<br><br>Current dimensions require NPLOSR$\leq$10. |

RANDOM OCCURRENCE DATA CARD SET Card No. 3

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| TCRIT | seconds | F10.4 | 1-10 | Critical time for laser designation (i.e., the length of the time interval immediately prior to impact during which the target must be continuously lased in order to have a successful Copperhead mission). |

RANDOM OCCURRENCE DATA CARD SET          Card No. 4

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| RNGPLS (1) | meters | F10.4 | 1-10 | RNGPLS (I) is the upper limit of the *Ith range* class for random occurrence line of sight data. |
| RNGPLS (2) | meters | F10.4 | 11-20 | |
| ⋮ | ⋮ | ⋮ | ⋮ | Range class 1 extends from zero to RNGPLS (1); range class I for I > 1 extends from RNGPLS (I) to RNGPLS (I+1). |
| RNGPLS (8) | meters | F10.4 | 71-80 | RNGPLS(I) < RNGPLS (I+1) |
| | | | | If NPLOSR > 8, then a second card must be used to contain RNGPLS (9) through RNGPLS (NPLOSR). |

RANDOM OCCURRENCE DATA CARD SET Card No. 5

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PRBLOS (1) | | F10.4 | 1-10 | PRBLOS (I) is the probability that a single vehicle in range class I is in view of the DO for the final TCRIT seconds of the Copperhead flight. |
| PRBLOS (2) | | F10.4 | 11-20 | |
| ⋮ | | ⋮ | | If NPLOSR >8, then a second card is needed to contain PRBLOS(9) through PRBLOS(NPLOSR). |
| PRBLOS (8) | | F10.4 | 71-80 | |

6. PE Data Block Input.

When reading PE data in PREPMS for transfer to TAPE 11, the first card (which indicates that a PE data record is to be read) must be of the form:

NEW RECORD, record name, 6

or

NEW REC, record name, 6

where record name is the name of the PE data record to be read (see section 5.4 for PE record names).

When using the TEMPORARY option with COPE, the PE data are headed by a card of the form:

TEMPORARY, PEDATA

or

TEMP, PEDATA

In either case (whether reading PE data in PREPMS or COPE), there is an optional card that can be inserted next:

USE TAPE 4

If this card is used, then the program (PREPMS or COPE) reads the remaining cards of the PE data from I/O unit 4 instead of the normal I/O unit 5; if this card is not used, then the PE data are read from I/O unit 5. In either case (whether I/O unit 4 or 5 is used), the remaining PE data cards are as described after this note:

Note: Because the PAM program (see Chapter 16) can write the PE data records directly to TAPE 11 without any need to run PREPMS, it is very unlikely that the need will arise to use PREPMS (or COPE) to read PE data as described in this section; however, the capability is provided.

PE DATA CARD SET Card No. 2a - 2dd

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| INDEX (I,J,1) | | I5 | 1-5 | INDEX (I,J,K) = 10*(I-1)+2*(J-1)+K where K = 1,2; J = 1,2,...,5; I = 1,2,...,6 |
| INDEX (I,J,2) | | I5 | 6-10 | The values are read two to a card and the values are arranged varying K first, then J, and then I. This procedure yields a set of 30 cards. The Nth such card having the values (2N-1) and 2N on it in 2I5 format. The index array is used to access the PETBL array. |

PE DATA CARD SET

Card No. 3a - 3b[24]
(600 cards)

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PETBL (I,J,1) | | F10.4 | 1-10 | PETBL (I,J,K) is the probability of engagement value for DO-to-target range K, visibility range J, and index value I. |
| PETBL (I,J,2) | | F10.4 | 11-20 | |
| PETBL (I,J,3) | | F10.4 | 21-30 | The values are input seven per card so that K varies first, then J, and finally I. |
| PETBL (I,J,4) | | F10.4 | 31-40 | K = 1,2,...,7<br>J = 1,2,...,10<br>I = 1,2,...60. |
| PETBL (I,J,5) | | F10.4 | 41-50 | |
| PETBL (I,J,6) | | F10.4 | 51-60 | |
| PETBL (I,J,7) | | F10.4 | 61-70 | This makes for a total of 600 cards with seven entries per card in 7F10.4 format. |

7. Posture Data Block Input.

When reading posture data in PREPMS for transfer to TAPE 11, the first card (which indicates that a Target Posture data record is to be read) must be of the form:

NEW RECORD, record name, 7

or

NEW REC, record name, 7

where record name is the name of the target posture data to be read (see section 5.3.6 for target posture record names).

When using the TEMPORARY option with COPE, the target posture data are headed by a card of the form:

TEMPORARY, TARGET POSTURE DISTRIBUTION

or

TEMPORARY, TGTPSTDIST

or

TEMP, TARGET POSTURE DISTRIBUTION

or

TEMP, TGTPSTDIST

In either case (whether reading target posture data in PREPMS or COPE), the remaining cards of the target posture data are as follow:

POSTURE DATA CARD SET Card No. 2

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| NRNGPS | | I5 | 1-5 | Number of range classes used for target posture data.<br><br>Current dimensions require NRNGPS $\leq$ 10. |

POSTURE DATA CARD SET    Card No. 3

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| RNGPST (1) | meters | F10.4 | 1-10 | RNGPST (I) is the upper limit of the Ith range class for posture data. |
| RNGPST (2) | meters | F10.4 | 11-20 | |
| RNGPST (3) | meters | F10.4 | 21-30 | The first range class runs from zero to RNGPST (1); the Ith range class extends from RNGPST (I) to RNGPST (I+1). |
| ⋮ | ⋮ | ⋮ | ⋮ | |
| RNGPST (8) | meters | F10.4 | 71-80 | Note RNGPST (I)$<$RNGPST (I+1) Also, if NRNGPS$>$8, the a second card is required: for RNGPST (9) through RNGPST (NRNGPS). |

POSTURE DATA CARD SET Card No. 4a

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PSTTBL (1,1) | | F10.4 | 1-10 | PSTTBL (I,1) is the probability that the target is completely obscured when at DO-to-target range RNGPST (I). |
| PSTTBL (2,1) | | F10.4 | 11-20 | |
| ⋮ | | ⋮ | ⋮ | |
| PSTTBL (8,1) | | F10.4 | 71-80 | If NRNGPS > 8, then a second card is required for PSTTBL (9,1) through PSTTBL (NRNGPS,1). |

POSTURE DATA CARD SET    Card No. 4b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PSTTBL (1,2) | | F10.4 | 1-10 | PSTTBL (I,2) is the probability that the target is fully exposed when at DO-to-target range RNGPST (I). |
| PSTTBL (2,2) | | F10.4 | 11-20 | |
| ' | | ' | ' | |
| ' | | ' | ' | |
| ' | | ' | ' | As usual, if NRNGPS> 8, a second card is needed for PSTTBL (9,2) through PSTTBL (NRNGPS,2). |
| PSTTBL (8,2) | | F10.4 | 71-80 | |

POSTURE DATA CARD SET Card No. 4c

| Variable name | UNITS | Format | columns | Description |
| --- | --- | --- | --- | --- |
| PSTTBL (1,3) | | F10.4 | 1-10 | PSTTBL (I,3) is the probability that the target is hull defilade (turret exposed) when at a DO-to-target range of RNGPST (I). |
| PSTTBL (2,3) | | F10.4 | 11-20 | |
| ⋮ | | ⋮ | ⋮ | |
| PSTTBL (8,3) | | F10.4 | 71-80 | Again, if NRNGPS > 8, a second card is needed for PSTTBL (9,3) through PSTTBL (NRNGPS,3). |

8. Invariant Data Block.

When reading invariant data in PREPMS for transfer to TAPE 11, the first card (which indicates that an invariant data record is to be read) must be of the form:

*NEW RECORD, record name, 8*

or

NEW REC, *record name*, 8

where record name is the name of the invariant data record to be read (see section 5.3.7 for invariant data record names).

When using the TEMPORARY option with COPE, the invariant data is headed by a card of the form:

TEMPORARY, INVARIANT

or

TEMP, INVARIANT

In either case (whether reading invariant data in PREPMS or COPE), the remaining cards of the invariant data are as follow:

INVARIANT DATA CARD SET Card No. 2

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| NRNGTT | | I5 | 1-5 | The number of range classes used with the LDWSS probability of hit numbers.<br><br>NRNGTT≤ 20 with current dimensions. |

INVARIANT DATA CARD SET Card No. 3a - 3t

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| RNGTTF (J) | meters | F10.4 | 1-10 | RNGTTF (J) is the Ith range at which probability of hit data is entered. RNGTTF (1) = 0.0 and RNGTTF (I) < RNGTTF (I+1). |
| TTF (1,J,1) | | F10.4 | 11-20 | TTF (I,J,K) is the probability of hit at range J using designator type I against a target in posture K.<br>I = 1 for GLLD<br>I = 2 for MULE<br>I = 3 for LTD<br>K = 1 for fully exposed<br>K = 2 for hull defilade (turret exposed) |
| TTF (1,J,2) | | F10.4 | 21-30 | |
| TTF (2,J,1) | | F10.4 | 31-40 | |
| TTF (2,J,2) | | F10.4 | 41-50 | |
| TTF (3,J,1) | | F10.4 | 51-60 | |
| TTF (3,J,2) | | F10.4 | 61-70 | There will be NRNGTT cards of this type with each card containing the same data points as its predecessor but for the next J value.<br>There will be a maximum of 20 such cards. |

INVAIANT DATA CARD SET Card No. 4

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| DLTT (1) | seconds | F10.4 | 1-10 | DLTT (I) is the Ith response time delay value to be used in interpolating with respect to delay time in the PETBL array. |
| DLTT (2) | seconds | F10.4 | 11-20 | |
| DLTT (3) | seconds | F10.4 | 21-30 | DLTT (1) = 0.0 and DLTT (I) < DLTT (I+1). |
| DLTT (4) | seconds | F10.4 | 31-40 | |
| DLTT (5) | seconds | F10.4 | 41-50 | |
| DLTT (6) | seconds | F10.4 | 51-60 | |

INVARIANT DATA CARD SET Card No. 5a

| Variable name | UNITS | Format | columns | Description |
| --- | --- | --- | --- | --- |
| PKTBL (1,1) | | F10.4 | 1-10 | PKTBL (I,1) is the probability of kill given a Copperhead hit against target type I in a fully exposed posture, |
| PKTBL (2,1) | | F10.4 | 11-20 | |
| PKTBL (3,1) | | F10.4 | 21-30 | |
| PKTBL (4,1) | | F10.4 | 31-40 | |
| PKTBL (5,1) | | F10.4 | 41-50 | |
| PKTBL (6,1) | | F10.4 | 51-60 | |
| PKTBL (7,1) | | F10.4 | 61-70 | |

AD-A100 285 ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY ABERDEEN PROV--ETC F/G 19/1
COPPERHEAD OPERATIONAL PERFORMANCE EVALUATION (COPE): COMPUTER --ETC(U)
MAR 81 R S SANDMEYER

UNCLASSIFIED AMSAA-TR-318 NL

4 OF 6

AD A
00285

INVARIANT DATA CARD SET Card No. 5b

| Variable name | UNITS | Format | columns | Description |
|---|---|---|---|---|
| PKTBL (1,2) | | F10.4 | 1-10 | PKTBL (I,2) is the probability of kill given a Copperhead hit against target type I in a hull defilade (turret exposed) posture. |
| PKTBL (2,2) | | F10.4 | 11-20 | |
| PKTBL (3,2) | | F10.4 | 21-30 | |
| PKTBL (4,2) | | F10.4 | 31-40 | |
| PKTBL (5,2) | | F10.4 | 41-50 | |
| PKTBL (6,2) | | F10.4 | 51-60 | |
| PKTBL (7,2) | | F10.4 | 61-70 | |

14.4.3 REPORT RECORD LIST Directive. The third type of input to PREPMS is a line of the form:

REPORT RECORD LIST

or

REP REC LIST

Inserting this line (card or card image) in the PREPMS input will cause the program to print a list of the names of the records on TAPE 11 at the time the line is encountered.

The usual practice is to place a line of this type in the input after all of the option word records and data block records so that the list produced will include all of the records created by the current run.

An example of the list produced is shown in Appendix C.

14.4.4 INTERPRET PEDATA Directive. The fourth type of input to PREPMS is a line of the form:

INTERPRET PEDATA

Inserting a line of this form in the PREPMS input will cause the program to print a list of the names of the current PE data records on TAPE 11 as well as the values of the nine parameters corresponding to each such record (see section 5.4).

### 14.5 PREPMS Output.

PREPMS writes to two files, TAPE 11 and TAPE 6 (also called OUTPUT). TAPE 11 receives the data block records, the option word records, the record named OPTNNUMS (which is automatically maintained by PREPMS with no user attention), and the record index array which is automatically maintained by the CDC word addressable mass storage file facilities. TAPE 6 (usually a line printer) is a sequential file on which PREPMS echoes the inputs that create the data block records and option name records as well as prints the lists generated by INTERPRET PEDATA and REPORT RECORD LIST directives.

A sample of PREPMS output (TAPE 6) is given in Appendix C, so further discussion is not given here.

## 14.6 PREPMS Diagnostic Messages.

Since several of the subroutines used by PREPMS are similar to those used in COPE, any diagnostic messages not found in this section (which gives only the PREPMS-peculiar messages) should be given in Chapter 7.

### 14.6.1 Messages of Informational Nature Not Terminating Program.

1. **PREPMS** NON-FATAL WARNING! YOU HAVE SEQUENCE NUMBERS IN COLS. 73-78 OF TAPE5. THEY WILL BE REMOVED BY THE PROGRAM; HOWEVER, THE RESULTS MAY BE SUSPECT.

This message is generated in SEPRC2 and is caused when the computer detects editor sequence numbers in the INPUT file (see section 7.1).

This message does not stop the program; however, it is very probable that the wrong data has been used if this message is printed. Great care should be taken to assure that the OUTPUT shows the correct (intended) values and that no sequence numbers were taken as input.

Note that the sequence numbers are removed only from those lines entered in free format. All lines entered as formatted data will still have sequence numbers which may be read instead of the intended data (for columns 73-78).

To avoid this problem, one should always use an 80 character line length when editing the input file for PREPMS (i.e., F,CH=80 on CDC).

2. **PREPMS** THIS IS A CREATION RUN FOR RECORD OPTNNUMS.

This message is not really a diagnostic but merely indicates to the user that this run of PREPMS creates a new TAPE 11. If this message is not present, it means that the current PREPMS run is modifying an existing TAPE 11 rather than creating a new one.

### 14.6.2 STOPS With Messages in Day Files.

1. STOP IN PREPMS: ERROR NUMBER 1.

This stop occurs when the program has read an input card which should be the first card of one of the four input types of section 14.4, but does not recognize the card as one of those four types.

This error is most likely to happen when either the card is mispunched or the previous input set (especially if it was type 1 or type 2 input) has the wrong number of cards.

2. STOP IN PREPMS: ERROR NUMBER 2.

This error occurs when the third field of a type 1 card of a type 1 input is not a number. (That is, the option word record input type starts with a card that has a non-numeral for the value of NREAD. See section 14.4.1 for the correct form of this card type).

3. STOP IN PREPMS: ERROR NUMBER 3.

This error occurs when the fourth field of a type 1 card (the value of NOPT1) of a type 1 input is not a number. (See section 14.4.1 for the correct form of this card type).

4. STOP IN PREPMS: ERROR NUMBER 4.

This error occurs when the first field of a type 2 card of a type 1 input is not a number. Again, see section 14.4.1 for the correct form of this card.

5. STOP IN PREPMS: ERROR NUMBER 5.

This error occurs when the second field of a type 2 card of a type 1 input is not a number. See section 14.4.1 for the correct form of this card.

6. STOP IN PREPMS: ERROR NUMBER 6.

This error occurs when the third field of the first card of a type 2 input is not a number. See section 14.4.2 (subsections 1 through 8) for the proper first card form for this input type.

7. STOP IN PREPMS: PROGRAM TERMINATION.

This means that the program reached the end of a file on TAPE 5 (INPUT) without any errors detected. It is the message that accompanies a successful run, but of course it is no guarantee that the data are correct (only that they are formatted in a valid form).

8. STOP IN SEPRC2: ERROR NUMBER 1.

This stop occurs only if the number of fields on a free formatted input card is greater than 10 or less than or equal to 0.

CHAPTER 15

# 15. PRBLOS PREPROCESSOR

The PRBLOS preprocessor is used to calculate the probability that a target randomly located along a route of approach within a certain designator-to-target range band will be in the line-of-sight of the designator for at least the final critical seconds of the COPPERHEAD trajectory.

This chapter gives: (1) the mathematical derivation of the method for computing the probability mentioned above, (2) a description of the program, (3) a glossary of program variables, (4) description of inputs, and (5) description of outputs. A program listing along with sample inputs and the resulting outputs are given in Appendix B.

## 15.1 Method For Calculating PRBLOS.

For a particular designator-to-target range band (e.g., ranges 2000m to 3000m) define F and G to be the cumulative distribution functions for in-view and out-of-view segment lengths respectively. That is, the probability that a randomly selected LOS segment for the given range band is of length less than or equal to X is F(X); the probability that a randomly selected out-of-view segment in the given range band is of length less than or equal to X is G(X).

Let $t_c$ be the critical time; that is the time interval at the end of the COPPERHEAD trajectory during which the target must be designated in order for the COPPERHEAD to track and hit the target. Also, let $v_t$ be the target velocity (target velocity is assumed constant).

Then the history of a target's exposure can be thought of as a sequence of alternating in-view and out-of-view segments. For example, the sequence $f_1$, $g_1$, $f_2$, $g_2$, $f_3$, $g_3$----,$h_n$, where the $f_i$'s are in-view segment lengths forming a sample from a population of segment lengths distributed according to the cumulative distribution F, the $g_i$'s are out-of-view segment lengths forming a sample from a population of segment lengths distributed according to cumulative distribution G, and $h_n$ is the last segment which is either an $f_i$ or a $g_i$ depending on whether the final segment is in-view or out-of-view.

Then for a particular target's exposure (i.e., in-view, out-of-view) history, one can calculate its probability of being in-view as probability target is in-view = fraction of target's path in-view =

$$\frac{\sum_{i=1}^{N} f_i}{\sum_{i=1}^{N} f_i + \sum_{i=1}^{M} g_i}$$

where M is equal to either N or N-1 depending on whether the last segment is respectively out-of-view or in-view.

In our problem though, we do not want the probability that the target is in-view, but rather the probability that it is in-view and will remain in-view for the next $t_c$ time units. This means that the target must be in an in-view segment and be at least $t_c*v_t$ distance from the end of that segment. The probability that the target is in such a location is equal to the fraction of the target's path that consists of points that are in-view and at least a distance $t_c*v_t$ from the next out-of-view segment.

For each in-view segment either (1) $f_i \leq t_c*v_t$ in which case that in-view segment contributes nothing to the total path length satisfying the conditions, or (2) $f_i > t_c*v_t$ distance from the next out-of-view segment.

Therefore, we have:

probability target is in-view and at least $t_c*v_t$ distance from next out-of-view segment =

fraction of target's path that is in-view and at least $t_c*v_t$ distance from the next out-of-view segment =

$$\frac{\sum_{i=1}^{N} \max\{0, f_i - t_c*v_t\}}{\sum_{i=1}^{N} f_i + \sum_{i=1}^{M} g_i}$$

where M and N are as before and max $\{0, t_i - t_c*v_t\}$ is the length contributed by the ith in-view segment to the total path length that is in-view and at least $t_c*v_t$ away from the next out-of-view segment.

Note: max $\{0, f_i-t_c*v_t\}$ means that the ith term of the sum is to be the maximum of the two quantities 0 and $(f_i-t_c*v_t)$.

Let $S_i$ = max $\{0, f_i-t_c*v_t\}$ then the $S_i$'s can be obtained by sampling $f_i$'s from a population with cumulative distribution F and then applying the formula $S_i$ = max $\{0, f_i-t_c*v_t\}$ . This is equivalent to sampling from a population with cumulative distribution S where:

$$S(0) = F(t_c*v_t) \text{ and}$$

$$S(X) = F(x + t_c*v_t) \text{ for } x > 0.$$

Hence, our probability that the target is in-view and at least $t_c{*}v_t$ from the next out-of-view segment can be written:

$$\frac{\frac{1}{N}\sum_{i=1}^{N} \max\{0,\ f_i - t_c{*}v_t\}}{\frac{1}{N}\sum_{i=1}^{N} f_i + \frac{1}{N}\sum_{i=1}^{M} g_i}$$

Now to obtain an overall estimate of this probability (which we call PRBLOS) we can imagine that we let our target wander back and forth over the various paths in the given range band. As this is done, $N \to \infty$ (and also $M/N \to 1$) so that

$$\frac{1}{N}\sum_{i=1}^{N} f_i \to M_f \text{ (mean of } f_i \text{ population)}$$

$$\frac{1}{N}\sum_{i=1}^{M} g_i \to \frac{M}{N}\left\{\frac{1}{M}\sum_{i=1}^{M} g_i\right\} \to \frac{M}{N} M_g \to M_g \text{ (mean of } g_i \text{ population)}$$

$$\frac{1}{N}\sum_{i=1}^{N} \max\{0,\ f_i - t_c{*}v_t\} = \frac{1}{N}\sum_{i=1}^{N} s_i \to M_s \text{ (mean of } s_i \text{ population)}$$

Therefore,

$$\text{PRBLOS} = \frac{M_s}{M_f + M_g}$$

is used as the probability that a vehicle randomly located in a given range band is in-view (from the designator's position) and will remain in-view for at least $t_c$ time units.

## 15.2 Description of PRBLOS Program and Glossary of Variables.

The program is extremely simple and consists of nothing but the steps required to perform the calculations described in 15.1 as well as to read the input and print the results.

The program can be summarized as four main steps:

(1) Read the input for the next case and print it for checking purposes; if no input, stop.

(2) Form the modified distribution (the one with cumulative distribution function S in section 15.1) and print it.

(3) Calculate the means of the three distributions F, G, and S (i.e., $M_f$, $M_g$, $M_s$) and from that calculate PRBLOS.

(4) Print PRBLOS and return to step (1) to see whether there is another case.

The subroutine AVEL of the program computes the mean of a distribution with cumulative distribution function F by using the formula:

$$M_f = \int_L^U xf(x)dx = UF(U)-LF(L) - \int_L^U F(x)dx$$

where f is the probability density function (i.e., f = F' almost everywhere) and U and L are respectively the upper and lower bounds of the distribution (in our cases, $0 \leq L < U < \infty$).

Since F is defined in the program by an array of ordered pairs ($x_i$, $F_i$) i = 1, 2...K where $F_i$ = probability that x is less than or equal to $x_i$, it is possible to approximate the right most integral by:

$$(*) \quad \int_L^U F(x)dx \simeq \frac{1}{2}\sum_{i=1}^{K-1} (x_{i+1}-x_i)(F_{i+1}+F_i).$$

Hence, the subroutine AVEL calculates the mean of a population with cumulative distribution function F using the formula:

$$M_f = x_K F_K - x_1F_1 - \frac{1}{2}\sum_{i=1}^{K-1} (x_{i+1}-x_i)(F_{i+1}+F_i).$$

The actual variable names used in the program are different from those used in sections 15.1 and 15.2; however, they are defined in the variable glossary so understanding should be fairly easy.

## GLOSSARY OF PRBLOS VARIABLES

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| A(6) | | (LOCAL) | | The A array contains up to 60 alphamuneric characters used as a case title |
| AVE | | (LOCAL) (F.P.) | meters | AVE is the value returned by subroutine AVEL as the mean length of the distribution with which it (AVEL) was called |
| AVEIV | | (LOCAL) | meters | AVEIV is the mean length of the in-view segment lengths ($M_f$ of section 15.1) |
| AVEIVM | | (LOCAL) | meters | AVEIVM is the mean length of the modified in-view segment lengths ($M_s$ of section 15.1) |
| AVEOOV | | (LOCAL) | meters | AVEOOV is the mean length of the out-of-view segment lengths ($M_g$ of section 15.1) |
| CD | | (LOCAL) | meters | CD is the critical distance; that is, the distance that the target will cover during the critical time ($t_c * v_t$ in the notation of section 15.1) |
| I | | (LOCAL) | | I is used as a DO-LOOP control variable and as a subscript value. |
| J | | (LOCAL) | | J is used as a DO-LOOP control variable and as a subscript value. |
| K | | (LOCAL) | | K is used as a DO-LOOP control variable and as a subscript value. |
| NIV | | (LOCAL) | | NIV is the number of points (ie, pairs, PRIV(I,1) and PRIV(I,2) used in the current case to describe the cumulative in-view segment length distribution (NIV $\leq$ 20 with current program dimensions) |
| NIVM | | (LOCAL) | | NIVM is the number of points (ie, pairs PRIVM(I,1) and PRIVM(I,2) used in the current case to describe the cumulative modified in-view segment length distribution (NIVM $\leq$ 20 with current program dimensions). |
| NOOV | | (LOCAL) | | NOOV is the number of points (ie, pairs PROOV(I,1) and PROOV(I,2) used in the current case to describe the cumulative out-of-view segment length distribution (NOOV $\leq$ 20 with current program dimensions). |
| NU | | (LOCAL) | | NU is equal to one less then the number of points used to describe the cumulative distribution whose mean is being calculated by subroutine AVEL. It is the number of terms in the sum of equation (*) of section 15.2. |
| N1 | | (LOCAL) (F.P.) | | N1 is the first dimension of the P array. In this program, N1 is always equal to 20. |

GLOSSARY OF PRBLOS VARIABLES - CONTINUED

| VARIABLE | TYPE | COMMON BLOCK | UNITS | DEFINITION |
|---|---|---|---|---|
| N2 | | (LOCAL) (F.P.) | | N2 is the second dimension of the P array. *In this program*, N2 is always equal to 2. |
| N3 | | (LOCAL) (F.P.) | | N3 is *the number of* points used to describe the cumulative distribution whose mean is being calculated by subroutine AVEL. |
| P(N1,N2) | | (LOCAL) (F.P.) | P(I,2) is in meters | P(I,1) is the probability that a segment length (randomly selected from the population whose mean is now being calculated by subroutine AVEL) is less than or equal to P(I,2). (where I=1, 2, ..., N3). |
| PRBIVM | | (LOCAL) | | This is the variable PRBLOS of section 15.1 (Since the program card uses PRBLOS as the program name it was necessary to call the variable by another name) It is the probability that a vehicle randomly located along an approach path in the given range band is currently in view and will remain in view for at least $t_c$ seconds. |
| PRIV(20,2) | | (LOCAL) | PRIV (I,2) is in meters | PRIV is the array describing the cumulative distribution of in-view segment lengths (F of section 15.1) PRIV (I,1) is the probability that a randomly chosen in-view segment is of length PRIV(I,2) or less. |
| PRIVM(20,2) | | (LOCAL) | PRIVM(I,2) is in meters | PRIVM is the array describing the cumulative distribution of modified in-view segment lengths (S of section 15.1). PRIVM(I,1) is the probability that a randomly chosen modified in-view segment is of length PRIVM(I,2) or less. |
| PROOV(20,2) | | (LOCAL) | PROOV(I,2) is in meters | PROOV is the array describing the cumulative distribution of out-of-view segment lengths (G of section 15.1). PROOV(I,1) is the probability that a randomly chosen out-of-view segment is of length PROOV(I,2) or less. |
| SUM | | (LOCAL) | | SUM is the value of the summation of equation (*) of section 15.2. |
| TC | | (LOCAL) | seconds | TC is the critical time ($t_c$ of section 15.1) It is the duration of the interval at the end of the Copperhead tr jectory during which the target must be (nearly) continuously lased in order that the round can track and hit the target. |
| TGTVEL | | (LOCAL) | meters/sec. | TGTVEL ($v_t$ of section 15.1) is the velocity of the target. |

## 15.3 PRBLOS Inputs.

For each case that is to be run with the PRBLOS program, a set of input cards (or card images) composed of card types 1 through 5b must be prepared. Multiple cases can be run by simply stacking card sets for the various cases one after another. The first card of each case has room (columns 21 through 80) for an identifying alphanumeric string which is reproduced on output for easier case identification. The program runs all cases until it either runs out of input or encounters an illegal input value.

Since the normal use of the PRBLOS program is to calculate the values to fill the RANDOM OCCURRENCE Data Block of COPE (in particular the PRBLOS array), each case run of the PRBLOS program corresponds to a given terrain, target velocity, and range band combination. For consistency, the PRIV data for PRBLOS should be obtained from the same source as the SEGLOS data for COPE. (Indeed, there is no reason why for a given terrain the same values cannot be used for PRIV(I,1) as for SEGLOS (I,1) and for PRIV(I,2) as for SEGLOS (I,J) in the (J-1)th range band.)

TGTVEL, TC, A CARD SET CARD NO. 1

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| TGTVEL | meters/sec. | F10.4 | 1-10 | The velocity of the target |
| TC | seconds | F10.4 | 11-20 | The critical time |
| A(1) | | A10 | 21-30 | A(1) through A(6) can be used to hold up to 60 characters of case heading which are copied from this card to the first output line of each case. This has no effect on calculations but enables the user to label the cases. |
| A(2) | | A10 | 31-40 | |
| A(3) | | A10 | 41-50 | |
| A(4) | | A10 | 51-60 | |
| A(5) | | A10 | 61-70 | |
| A(6) | | A10 | 71-80 | |

NIV CARD SET CARD NO. 2

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| NIV | | I5 | 1-5 | This is the number of points to be read to describe the cumulative in-view segment length distribution.<br>NIV controls reading of PRIV:<br>PRIV (I,1) for I=1,2...,NIV<br>PRIV (I,2) for I=1,2...,NIV<br><br>NOTE: Current program dimensions require NIV $\leq$ 20. |

PRIV (I,1) CARD SET CARD NO. 3a

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| PRIV(1,1) | | F10.5 | 1-10 | PRIV(I,1) is the probability that a randomly chosen in-view segment has length less than or equal to PRIV(I,2). |
| PRIV(2,1) | | F10.5 | 11-20 | NOTE: |
| PRIV(3,1) | | F10.5 | 21-30 | PRIV(I,1) $\leq$ PRIV(I+1,1) for I = 1,2, NIV-1; |
| PRIV(4,1) | | F10.5 | 31-40 | 0 $\leq$ PRIV(I,1) $\leq$ 1 for I = 1,2...,NIV; and |
| PRIV(5,1) | | F10.5 | 41-50 | PRIV(NIV,1) = 1.0 |
| PRIV(6,1) | | F10.5 | 51-60 | If NIV > 8, then an additional card (or two) is needed for PRIV(9,1) through PRIV(NIV,1) |
| PRIV(7,1) | | F10.5 | 61-70 | The additional card(s) should have 8F10.4 format also. |
| PRIV(8,1) | | F10.5 | 71-80 | PRIV(I,1) and PRIV(I,2) together correspond to the $F_i$ and $X_i$ values respectively (in the notation of section 15.2) for the in-view distribution. |

PRIV (I,2) CARD SET CARD NO. 3b

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| PRIV(1,2) | meters | F10.5 | 1-10 | PRIV(I,2) is the in-view segment length corresponding to cumulative probability PRIV(I,1) |
| PRIV(2,2) | meters | F10.5 | 11-20 | |
| PRIV(3,2) | meters | F10.5 | 21-30 | NOTE: PRIV(I,2) $\leq$ PRIV(I+1,2) for I = 1,2, ...,NIV-1 |
| PRIV(4,2) | meters | F10.5 | 31-40 | |
| PRIV(5,2) | meters | F10.5 | 41-50 | |
| PRIV(6,2) | meters | F10.5 | 51-60 | If NIV>8, then an additional card (or two) is needed for PRIV(9,2) through PRIV(NIV,2). The additional card(s) have 8F10.4 format. |
| PRIV(7,2) | meters | F10.5 | 61-70 | |
| PRIV(8,2) | meters | F10.5 | 71-80 | |

NOOV CARD SET CARD NO. 4

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
| --- | --- | --- | --- | --- |
| NOOV | | I5 | 1-5 | NOOV is the number of points to be read to describe the cumulative out-of-view segment length distribution.<br>NOOV controls reading of PROOV:<br>PROOV(I,1) for I=1,2...,NOOV<br>PROOV(I,2) for I=1,2...,NOOV<br>NOTE: Current dimensions require NOOV $\leq$ 20. |

PROOV (I,1) CARD SET CARD NO. 5a

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| PROOV(1,1) | | F10.4 | 1-10 | PROOV(I,1) is the probability that a randomly chosen out-of-view segment has length less than or equal to PROOV(I,2). |
| PROOV(2,1) | | F10.4 | 11-20 | |
| PROOV(3,1) | | F10.4 | 21-30 | NOTE: PROOV(I,1) ≤ PROOV (I+1,1) for I = 1,2,...,NOOV-1; 0 ≤ PROOV(I,1) ≤ 1.0 for I = 1,2..., NOOV; and PROOV (NOOV,1) = 1.0 |
| PROOV(4,1) | | F10.4 | 31-40 | |
| PROOV(5,1) | | F10.4 | 41-50 | |
| PROOV(6,1) | | F10.4 | 51-60 | |
| PROOV(7,1) | | F10.4 | 61-70 | If NOOV > 8, then an additional card(or two) is needed for PROOV(9,1) through PROOV(NOOV,1). The additional card(s) should have 8F10.4 format. |
| PROOV(8,1) | | F10.4 | 71-80 | PROOV(I,1) and PROOV(I,2) correspond to a point on the cumulative distribution G of section 15.1 (a point with coordinates (PROOV(I,2),PROOV(I,1)) in the usual notation. |

PROOV(I,2) CARD SET CARD NO. 5b

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| PROOV(1,2) | meters | F10.4 | 1-10 | PROOV(I,2) is the out-of-view segment length corresponding to cumulative probability PROOV (I,1). |
| PROOV(2,2) | meters | F10.4 | 11-20 | |
| PROOV(3,2) | meters | F10.4 | 21-30 | NOTE: |
| PROOV(4,2) | meters | F10.4 | 31-40 | PROOV(I,2) $\leq$ PROOV (I+1,2) for I = 1,2,...,NOOV-1; |
| PROOV(5,2) | meters | F10.4 | 41-50 | 0.0 $\leq$ PROOV(I,2) $\leq$ 1.0 for I = 1, 2,...,NOOV. |
| PROOV(6,2) | meters | F10.4 | 51-60 | |
| PROOV(7,2) | meters | F10.4 | 61-70 | If NOOV > 8, then an additional card (or two) is needed for PROOV(9,2) through PROOV(NOOV,2). The additional card(s) have 8F10.4 format. |
| PROOV(8,2) | meters | F10.4 | 71-80 | |

## 15.4 PRBLOS Output and Diagnostics.

15.4.1 Output. The output of the PRBLOS program is very brief and easy to understand.

For each case run, the first line of output prints the case identifier (if any). The next line prints the target velocity and critical time values. Then comes the PRIV array preceded by NIV. Next the PROOV array preceded by NOOV and the PRIVM array preceded by NIVM. Finally, it prints the average segment lengths of the three distributions and the PRBLOS value followed by the words "END OF CASE".

In the special case that $t_c*v_t$ is greater than the longest possible in-view segment, a message to that effect is printed along with the value of PRBLOS (namely, 0.0). In this case, the PRIVM array is not printed.

Note that PRBIVM is labeled PRBLOS when printed. (This is because PRBIVM is a second choice name for PRBLOS used only because the program card name PRBLOS prevented the use of a variable PRBLOS in the program).

A program listing as well as sample input and output for PRBLOS is contained in Appendix B.

### 15.4.2 Diagnostic ("STOP") Messages in Dayfile.

1. STOP END OF INPUT.

This message means that the program has encountered the end of file of TAPE5 (INPUT). This is the normal program termination message, but of course, it is no guarantee that the results are correct.

2. STOP NIV>20 EXCEEDS DIMENSIONS OF PRIV ARRAY.

This message means the user attempted to read in an NIV value greater than 20. It may occur because NIV is not right justified in columns 1-5 of its input card. Of course, it may also occur if NIV is really greater than 20; if this is the case, then the user must either use fewer points to describe PRIV or redimension PRIV in the program (and change N1 in first call of AVEL. Redimensioning of PRIVM and changing N1 in the third call of AVEL may also be required).

3. STOP NOOV>20 EXCEEDS DIMENSIONS OF PROOV ARRAY.

This message means the user attempted to read in an NOOV value greater than 20. It may occur because NOOV is not right justified in columns 1-5 of its input card. Of course, it may also occur if NIV is really greater than 20; if this is the case, then the user must either use fewer points to describe PROOV or redimension PROOV in the program (and change N2 in the second call of AVEL).

 Next page is blank.

CHAPTER 16

# 16. PAM PREPROCESSOR

Unlike the other programs documented in this report (COPE, PREPMS, and PRBLOS), the PAM program was not written by the author of this report. Since the author (Michael Starks, GWD, AMSAA) of PAM will document it in a separate report, this chapter is limited to a brief explanation of the program's purpose and a description of the input. Appendix A contains a program listing as well as sample inputs and outputs.

## 16.1 General Description of PE Program (PAM).

The PAM program is used to compute the probability of engagement of the target by the COPPERHEAD seeker. More precisely, probability of engagement (PE) is the probability that the COPPERHEAD seeker receives enough reflected laser energy from the target for the COPPERHEAD round to initiate maneuver toward the target and that the target is within the maneuver footprint of the COPPERHEAD.

The PAM model produces a table of 4200 PE values (6 cloud ceiling altitudes x 10 meteorological visibility limits x 5 delay times x 7 designator-to-target ranges x 2 "MUTS" factors). (The multi-unit target factor ("MUTS" factor) allows TLE to be reduced when the predicted impact point is bracketed by a column of target vehicles.)

Such a table then has PE values as a function of cloud ceiling, visibility limit, delay time, MUTS factor, and designator-to-target range. In addition, the set of PE's in the table is a function of the 9 parameters mentioned in section 5.4 (namely, nominal response time, designator type (power), target velocity, gun-target range, reflectivity of target, angle T, deflection bias (distance between footprint centroid and target path point of closest approach), target heading, and seeker sensitivity). Hence, for each different combination of values of the 9 parameters a different set of PE values is needed which, in turn, requires another run of the PAM program.

The PAM program includes the code mentioned in section 5.4 that generates PE data block record names from the values of the 9 parameters. The subroutine PENAM2 is included in PAM and is identical to the PENAME subroutine in COPE and PREPMS except for the change in name and the fact that the data statement for the XVALUE array is included in PENAM2 rather than in a separate BLOCK DATA subprogram as in COPE and PREPMS. The error stops with messages produced by PENAM2 correspond to those for PENAME as explained in section 7.5.

Assuming that the user runs PAM with values of the 9 parameters that are permitted by PENAME (see section 5.4 and discussion of XVALUE array), then a PE DATA record name is generated. The PE values are loaded in the arrays PE (for writing to TAPE 11 as a PE data record for use by COPE) and PENG (for writing to TAPE 6 (output) to provide a hard copy of the PE values).

In order that the values written to TAPE 11 are accessible by COPE, of course, it is necessary to catalog TAPE 11 and use it with the COPE runs. The interface between PAM and COPE has been covered in section 5.4.

The output of PAM consists mainly of the inputs which are echoed for checking purposes (some of the angles are converted to radians before being printed). The PE values are printed in 70 groups of 60 (6 cloud ceiling altitudes, 10 visibility limits) with each group corresponding to a particular designator-to-target range, time delay, and MUTS factor combination.

A program listing together with sample inputs and outputs is included in in Appendix A.

## 16.2 PAM Inputs.

The PAM program takes all its inputs in free format (in CDC terminology, it uses list-directed reads). This means that on each card in the following description that takes more than one variable, the values of the variables are punched one after another with commas between successive values. (See sample input in Appendix A).

IMF CARD SET CARD NO. 1

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| IMF | | Free | | Mission type flag<br>IMF = 1 for preplanned target (priority of not)<br>IMF = 2 for target of opportunity |

ETH CARD SET CARD NO. 2

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| ETH | Joules/km$^2$ | Free | | ETH is the energy threshold. That is, the amount of energy/area that must reach the seeker to initiate maneuver. |

AOF CARD SET CARD NO. 3

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| AOF | Degrees | Free | | AOF is the angle of fall of the COPPERHEAD round (measured from the horizontal). |

TH,PCA CARD SET CARD NO. 4

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| TH | Degrees | Free | | Target heading. The angle between the gun-to-target line and the target path when measured counter-clockwise from the gun-target line. |
| PCA | Degrees | Free | | Point of closest approach (deflection bias). This is the distance between the footprint centroid and the closest point on the target's path to the centroid. |

AZDT, V, RHO, ED, TR CARD SET CARD NO. 5

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| AZDT | Degrees | Free | | AZDT is the angle T (that is, the angle between the designator-target line and gun-target line measured counter-clockwise from the gun-target line.)* |
| V | Meters/Sec | Free | | V is the target velocity. |
| RHO | | Free | | RHO is the target's reflectivity. |
| ED | Millijoules | Free | | Designator energy. |
| TR | Seconds | Free | | Nominal respone time (time from DO's call for fire to expected round arrival. It includes time-of-flight in this case. |
| | | | | **Note: AZDT is the nominal angle* T; that is, the angle between the designator-target line and the gun target-line when TR seconds have passed since DO's call for fire. |

NK, RNG, ACCX, ACCY CARD SET CARD NO. 6

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
| --- | --- | --- | --- | --- |
| NK | | Free | | NK is the Monte Carlo sample size used in the program. |
| RNG | Meters | Free | | Gun-to-target range (must be the range for which footprints are supplied). |
| ACCX | Meters | Free | | ACCX is the COPPERHEAD total delivery error standard deviation in the deflection direction.* |
| ACCY | Meters | Free | | ACCY is the COPPERHEAD total delivery error standard deviation in the range direction.* |
| | | | | *Note: These are the delivery error standard deviations for unguided COPPERHEAD. |

IDRMN, IDRMX, IVMX CARD SET CARD NO. 7

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| IDRMN | Km | Free | | Minimum designator-to-target range. |
| IDRMX | Km | Free | | Maximum designator-to-target range. |
| IVMX | Km | Free | | Maximum visibility range. |

NI CARD SET CARD NO. 8a

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| NI(J) | | Free | | NI(J) is the number of points used to describe the maneuver footprint **for the Jth altitude.**<br><br>J = 1 for highest altitude<br>2 for second highest altitude<br>.<br>.<br>.<br>6 for lowest altitude<br><br>The six altitudes for which footprints are entered are the six altitudes used for cloud ceiling altitudes in COPE weather data.<br><br>Note that there will be a set of cards 8a, 8b, 8c for each footprint for a total of six such sets in all. They must be input in order of ascending J's.<br><br>With current dimensions NI(J) $\leq$ 11 for all J. |

THEMH CARD SET CARD NO. 8b

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| THEMH(1,J) | degrees | free | | THEMH(I,J) is the angle used to represent the Ith point on the boundary of the Jth footprint. |
| THEMH(2,J) | degrees | free | | |
| THEMH(3,J) | degrees | free | | Note that the Jth footprint is represented by a set of points$\{(\rho_i, \theta_i)\}$ in polar coordinates, where |
| THEMH(4,J) | degrees | free | | $\theta_i$ = THEMH(I,J) |
| THEMH(5,J) | degrees | free | | $\rho_i$ = DISMH(I,J). |
| THEMH(6,J) | degrees | free | | THEMH(1,J) = 0.0 |
| THEMH(7,J) | degrees | free | | THEMH(NI(J), J) = 180.0 |
| THEMH(8,J) | degrees | free | | The angles are measured off the ray in the positive range direction from the centroid of the footprint. |
| THEMH(9,J) | degrees | free | | |
| THEMH(10,J) | degrees | free | | |
| THEMH(11,J) | degrees | free | | Because the footprint is symmetric about the range axis, only one half of it is read in. It is for this reason that the THEMH(I,J) values do not exceed 180 degrees. |
| | | | | Also note that THEMH(I,J) < THEMH(I+1,J) for I = 1,2,...,NI(J) - 1 |
| | | | | If NI(J) < 11, then only NI(J) entries occur on this card rather than the 11 entries shown. |

DISMH CARD SET CARD NO. 8c

| VARIABLE NAME | UNITS | FORMAT | COLUMNS | DESCRIPTION |
|---|---|---|---|---|
| DISMH(1,J) | meters | free | | DISMH(I,J) is the distance from the centroid to the Ith point on the boundary of the Jth footprint. |
| DISMH(2,J) | meters | free | | |
| DISMH(3,J) | meters | free | | |
| DISMH(4,J) | meters | free | | |
| DISMH(5,J) | meters | free | | |
| DISMH(6,J) | meters | free | | |
| DISMH(7,J) | meters | free | | |
| DISMH(8,J) | meters | free | | |
| DISMH(9,J) | meters | free | | |
| DISMH(10,J) | meters | free | | |
| DISMH(11,J) | meters | free | | If NI(J) < 11, then this card will have only NI(J) entries rather than the 11 shown. |

# APPENDIX A

## LISTING AND SAMPLE CASE OF PAM

# APPENDIX A

## LISTING AND SAMPLE CASE OF PAM

This appendix contains:

(1) A listing of the FORTRAN code for the PAM program. In addition to the routines in this listing, PAM also uses the function URAN31 which is included in the COPE listing in Appendix D.

(2) Three sets of sample case input for PAM.

(3) Three sample runstreams for PAM (one corresponding to each set of sample input).

(4) Sample output created by running PAM with the first set of sample input.

Note: The column numbers included in some of the listings in this appendix are not a part of the program code, the data, or the output, but are provided only for the reader's convenience.

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      PROGRAM PAM (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7,TAPE11)  037810  PR-PAM
C                                                                       037820  PR-PAM
C     **  THIS VERSION OF PAM IS CONFIGURED FOR USE AS A PREPROCESSOR   037830  PR-PAM
C     **  FOR CORE.  MAKE THESE CHANGES FOR STAND-ALONE RUNS:           037840  PR-PAM
C     **     1.  DELETE CALL TO OPENMS (FIRST EXECUTABLE STATEMENT)     037850  PR-PAM
C     **     2.  DELETE CALL TO PENAM2                                  037860  PR-PAM
C     **     3.  DELETE CALLS TO WRITMS AND CLOSMS                      037870  PR-PAM
C     **     4.  DELETE SUBROUTINE PENAM2                               037880  PR-PAM
C                                                                       037890  PR-PAM
      DIMENSION ACQ(6,6), H(6), TOT(6), INDEX(6,5,2), PE(60,10,7),      037900  PR-PAM
     1  AMTX(4260), INDX11(2001), THEMH(11), TOT2(6,10), DISMH(11,6),   037910  PR-PAM
     2  DISMNI(11), THEMNI(11,6), NI(6), LKUP(7,5,2), PENG(70,10,6),    037920  PR-PAM
     3  GAMAPY(11), DTS(5)                                              037930  PR-PAM
C                                                                       037940  PR-PAM
      EQUIVALENCE (AMTX(1),INDEX(1)), (AMTX(61),PE(1))                  037950  PR-PAM
C                                                                       037960  PR-PAM
      REAL MDIS                                                         037970  PR-PAM
C                                                                       037980  PR-PAM
C     GAMAPY HOLDS THE ATTENUATION COEFFICIENTS AS A FUNCTION OF RANGE  037990  PR-PAM
      DATA GAMAPY /2.6644, 1.2058, .7496, .5317, .4059, .3252, .2727, .2038000  PR-PAM
     1334, .2030, .1803, .1618/                                         038010  PR-PAM
      PI=3.1416                                                         038020  PR-PAM
C                                                                       038030  PR-PAM
C     H ARRAY HOLDS ALTITUDES FOR WHICH ACQUISITION IS CHECKED          038040  PR-PAM
      DATA H /1372.0, 914.0, 762.0, 610.0, 457.0, 304.0/                038050  PR-PAM
C                                                                       038060  PR-PAM
C      DTS ARRAY HOLDS DELAY TIMES PLAYED                               038070  PR-PAM
      DATA DTS /0.0, 30., 90., 150., 300./                              038080  PR-PAM
      DATA IRN1 /1234567/                                               038090  PR-PAM
      DATA IRN2 /7654321/                                               038100  PR-PAM
      CALL OPENMS (11,INDX11,2001,1)                                    038110  PR-PAM
      READ (5,*) INF                                                    038120  PR-PAM
      IF (INF.EQ.1) WRITE (6,300)                                       038130  PR-PAM
      IF (INF.EQ.2) WRITE (6,310)                                       038140  PR-PAM
      READ (5,320) ETH                                                  038150  PR-PAM
      WRITE (6,330) ETH                                                 038160  PR-PAM
      SKSEN=ETH*1000000.                                                038170  PR-PAM
      READ (5,*) ADF                                                    038180  PR-PAM
      ADF=ADF*.01745329252                                              038190  PR-PAM
      READ (5,*) TH,PCA                                                 038200  PR-PAM
      READ (5,*) AZDT,V,PHO,ED,TP                                       038210  PR-PAM
      WRITE (6,350) AZDT,V,PHO,ED,TP,ADF                                038220  PR-PAM
      WRITE (6,340) TH,PCA                                              038230  PR-PAM
CSEC  THE CONSTANT VALUES TO WHICH ED IS COMPARED IN THE NEXT TWO       038240  PR-PAM
CSEC  STATEMENTS ARE FALSE VALUES INSERTED TO MAKE THIS LISTING         038250  PR-PAM
CSEC  UNCLASSIFIED.  THEY SHOULD BE CHANGED TO THE CORRECT VALUES WHEN  038260  PR-PAM
CSEC  USING THE PROGRAM IN A SECURE ENVIRONMENT.  THESE TWO VALUES      038270  PR-PAM
CSEC  ARE RESPECTIVELY THE DESIGNATOR ENERGIES FOR GLLD AND MULE.       038280  PR-PAM
CSEC  THE VALUE FOR LTD IS EQUAL TO THAT FOR MULE.                      038290  PR-PAM
CSEC  THE FALSE VALUES IN THIS LISTING ARE USED ONLY FOR SAMPLE CASES.  038300  PR-PAM
CSEC  THE PHYSICAL UNITS USED FOR THESE VALUES ARE THE SAME AS FOR THE  038310  PR-PAM
CSEC  VARIABLE ED.                                                      038320  PR-PAM
      IF (ED.EQ..20) IDSG=1                                             038330  PR-PAM
      IF (ED.EQ..14) IDSG=2                                             038340  PR-PAM

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM

(PAGE 1 OF 11)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      READ (5,*) NK,RNG,ACCX,ACCY                                       038350   PR-PAM  56
      GTR=RNG/1000.                                                     038360   PR-PAM  57
      WRITE (6,360) NK,RNG,ACCX,ACCY                                    038370   PR-PAM  58
      READ (5,*) IDRMN,IDRMX,IVMX                                       038380   PR-PAM  59
      WRITE (6,370) IDRMN,IDRMX,IVMX                                    038390   PR-PAM  60
      DO 110 J=1,6                                                      038400   PR-PAM  61
      READ (5,*) NI(J)                                                  038410   PR-PAM  62
      NIJ=NI(J)                                                         038420   PR-PAM  63
      WRITE (6,390) J,NI(J)                                             038430   PR-PAM  64
      READ (5,*) (THEMN(I,J),I=1,NIJ)                                   038440   PR-PAM  65
      READ (5,*) (DISMN(I,J),I=1,NIJ)                                   038450   PR-PAM  66
      WRITE (6,380) (THEMN(I,J),I=1,NIJ)                                038460   PR-PAM  67
      WRITE (6,400) (DISMN(I,J),I=1,NIJ)                                038470   PR-PAM  68
      DO 100 I=1,NIJ                                                    038480   PR-PAM  69
      THEMN(I,J)=THEMN(I,J)*0.01745329252                               038490   PR-PAM  70
  100 CONTINUE                                                          038500   PR-PAM  71
  110 CONTINUE                                                          038510   PR-PAM  72
      CALL PENAM2 (TR,IDSG,V,GTR,RHO,AZDT,PCA,TH,SKSEN,IDCODE)          038520   PR-PAM  73
      WRITE (6,410) IDCODE,IDCODE                                       038530   PR-PAM  74
      WRITE (6,420) V,TR,ED,RHO                                         038540   PR-PAM  75
C                                                                       038550   PR-PAM  76
C                                                                       038560   PR-PAM  77
      TH=TH*.01745329252                                                038570   PR-PAM  78
      DO 240 IDR=IDRMN,IDRMX                                            038580   PR-PAM  79
      RD=FLOAT(IDR)                                                     038590   PR-PAM  80
      DO 230 IV=1,IVMX                                                  038600   PR-PAM  81
      DO 220 IDELT=1,5                                                  038610   PR-PAM  82
      DT=DTS(IDELT)                                                     038620   PR-PAM  83
C                                                                       038630   PR-PAM  84
      DO 210 IMTS=1,2                                                   038640   PR-PAM  85
C         COMPUTE SIGMA OF RANDOM TLE                                   038650   PR-PAM  86
      TLOC=SQRT(2500.+(TR+DT)**2)                                       038660   PR-PAM  87
      IF (IMF.EQ.1) TLOC=SQRT((2500.+1.5*((TR+DT)**2.)))                038670   PR-PAM  88
      IF (IMTS.EQ.2.AND.IDELT.EQ.1) TLOC=.68*TLOC                       038680   PR-PAM  89
C                                                                       038690   PR-PAM  90
C         COMPUTE COORDINATES OF POINT OF CLOSEST APPROACH              038700   PR-PAM  91
      PCAX=PCA*COS(TH)                                                  038710   PR-PAM  92
      PCAY=PCA*SIN(TH)                                                  038720   PR-PAM  93
C         COMPUTE TARGET COORDINATES FOR GIVEN DELAY/VELOCITY           038730   PR-PAM  94
      BIASX=PCAX+(SIN(TH)*V*DT)                                         038740   PR-PAM  95
      BIASY=PCAY-(COS(TH)*V*DT)                                         038750   PR-PAM  96
C                                                                       038760   PR-PAM  97
C         BEGIN MONTE CARLO LOOP BY SAMPLING TLE                        038770   PR-PAM  98
      DO 150 K=1,NK                                                     038780   PR-PAM  99
      CALL NRAN31 (SX1,SX2,IRN1)                                        038790   PR-PAM 100
      TLOCY=SX1*TLOC                                                    038800   PR-PAM 101
      TLOCX=SX2*TLOC                                                    038810   PR-PAM 102
C         SUM RANDOM ERROR TO BIAS ERROR                                038820   PR-PAM 103
      TLEX=TLOCX+BIASX                                                  038830   PR-PAM 104
      TLEY=TLOCY+BIASY                                                  038840   PR-PAM 105
C         SAMPLE BALLISTIC ACCURACY                                     038850   PR-PAM 106
      CALL NRAN31 (SX1,SX2,IRN2)                                        038860   PR-PAM 107
      BIRY=SX1*ACCY                                                     038870   PR-PAM 108
      BIRX=SX2*ACCX                                                     038880   PR-PAM 109

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM - CONT'D

(PAGE 2 OF 11)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

C                                                                         038890  PR-PAM 110
C           AQUISITION VOLUME ENERGY COMPARISON                           038900  PR-PAM 111
C                                                                         038910  PR-PAM 112
      DO 140 IC=1,6                                                       038920  PR-PAM 113
      DO 130 IA=IC,6                                                      038930  PR-PAM 114
      YINT=-H(IA)*COT(ADF)+BIPY                                           038940  PR-PAM 115
      XINT=BIPX                                                           038950  PR-PAM 116
      X=(TLEX-XINT)                                                       038960  PR-PAM 117
      Y=(TLEY-YINT)                                                       038970  PR-PAM 118
      ALPHA=ATAN2(X,Y)                                                    038980  PR-PAM 119
      ANGSUM=(AZDT*.0174532925 2)+ALPHA                                   038990  PR-PAM 120
      GAM=GAMARY(IV)                                                      039000  PR-PAM 121
      TD=EXP(-GAM*RD)                                                     039010  PR-PAM 122
      HS=(H(IA)*3.28)                                                     039020  PR-PAM 123
      RS=SQRT((XINT-TLEX)**2.+(YINT-TLEY)**2.+H(IA)**2.)                  039030  PR-PAM 124
      PS=RS/1000.0                                                        039040  PR-PAM 125
      TS=EXP((-GAM*RS)*(1.-EXP(-.00025*HS))/(.00025*HS))                  039050  PR-PAM 126
      TFACT=COS(ANGSUM)                                                   039060  PR-PAM 127
      ES=(ED*TD*TS*RHO*TFACT)/(PI*PS*PS)                                  039070  PR-PAM 128
      IF (ETH.GT.ES) GO TO 130                                            039080  PR-PAM 129
C                                                                         039090  PR-PAM 130
C           MANUEVERABILITY FOOTPRINT DISTANCE COMPARISON                 039100  PR-PAM 131
C                                                                         039110  PR-PAM 132
      TANG=ABS((TLEY-BIPY)/(TLEX-BIPX))                                   039120  PR-PAM 133
      BETA=ATAN(TANG)                                                     039130  PR-PAM 134
      BETA=1.5708-BETA                                                    039140  PR-PAM 135
      IF (TLEY.LT.BIPY) BETA=3.1416-BETA                                  039150  PR-PAM 136
      DIST=SQRT((TLEX-BIPX)**2.+(TLEY-BIPY)**2.)                          039160  PR-PAM 137
      NIJ=NI(IA)                                                          039170  PR-PAM 138
      DO 120 J=1,NIJ                                                      039180  PR-PAM 139
      THEMHI(J)=THEMH(J,IA)                                               039190  PR-PAM 140
      DISMHI(J)=DISMH(J,IA)                                               039200  PR-PAM 141
  120 CONTINUE                                                            039210  PR-PAM 142
      CALL DVDINT (BETA,MDIS,THEMHI,DISMHI,NIJ,2)                         039220  PR-PAM 143
      IF (DIST.GT.MDIS) GO TO 130                                         039230  PR-PAM 144
      ACQ(IC,IA)=ACQ(IC,IA)+1.0                                           039240  PR-PAM 145
      GO TO 140                                                           039250  PR-PAM 146
  130 CONTINUE                                                            039260  PR-PAM 147
  140 CONTINUE                                                            039270  PR-PAM 148
  150 CONTINUE                                                            039280  PR-PAM 149
C                                                                         039290  PR-PAM 150
C           ACCUMULATE RESULTS OVER CEILING AND ALTITUDE                  039300  PR-PAM 151
      DO 160 IC=1,6                                                       039310  PR-PAM 152
      DO 160 IA=1,6                                                       039320  PR-PAM 153
      ACQ(IC,IA)=ACQ(IC,IA)/FLOAT(NK)                                     039330  PR-PAM 154
  160 TOT(IC)=TOT(IC)+ACQ(IC,IA)                                          039340  PR-PAM 155
C     WRITE(6,102)(H(IC),IC=1,6),(H(IA),(ACQ(IC,IA),IC=1,6),IA=1,6)       039350  PR-PAM 156
C     WRITE(6,100) (TOT(IC),IC=1,6)                                       039360  PR-PAM 157
C                                                                         039370  PR-PAM 158
C           INDEX RESULTS FOR CORE INTERFACE AND FOR TAPE6 OUTPUT         039380  PR-PAM 159
C                                                                         039390  PR-PAM 160
      DO 170 II=1,6                                                       039400  PR-PAM 161
      JPE=10*(II-1)+2*(IDELT-1)+INTS                                      039410  PR-PAM 162
      INDEX(II,IDELT,INTS)=JPE                                            039420  PR-PAM 163

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM - CONT'D

(PAGE 3 OF 11)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

  170 PE(JPE,IV,IDR)=TOT(II)                                          039430  PP-PAM 164
      DO 180 II=1,6                                                   039440  PP-PAM 165
      IPF=10*(IDR-1)+2*(IDELT-1)+INTS                                 039450  PP-PAM 166
      LKUP(IDR,IDELT,INTS)=IPF                                        039460  PP-PAM 167
  180 PENG(IPE,IV,II)=TOT(II)                                         039470  PP-PAM 168
      DO 190 IC=1,6                                                   039480  PP-PAM 169
      TOT2(IC,IV)=TOT(IC)                                             039490  PP-PAM 170
  190 CONTINUE                                                        039500  PP-PAM 171
      DO 200 IC=1,6                                                   039510  PP-PAM 172
      DO 200 IA=1,6                                                   039520  PP-PAM 173
      ACQ(IC,IA)=0.                                                   039530  PP-PAM 174
      TOT(IC)=0.                                                      039540  PP-PAM 175
  200 CONTINUE                                                        039550  PP-PAM 176
  210 CONTINUE                                                        039560  PP-PAM 177
  220 CONTINUE                                                        039570  PP-PAM 178
  230 CONTINUE                                                        039580  PP-PAM 179
  240 CONTINUE                                                        039590  PP-PAM 180
      DO 250 I=1,7                                                    039600  PP-PAM 181
      DO 250 J=1,5                                                    039610  PP-PAM 182
      DO 250 K=1,2                                                    039620  PP-PAM 183
      IF ((K.EQ.2).AND.(J.GT.1)) GO TO 250                            039630  PP-PAM 184
      DT=DTS(J)                                                       039640  PP-PAM 185
      WRITE (6,430) I,DT,K                                            039650  PP-PAM 186
      IQ=10*(I-1)+2*(J-1)+K                                           039660  PP-PAM 187
      WRITE (6,440) ((PENG(IQ,JQ,KQ),KQ=1,6),JQ=1,10)                 039670  PP-PAM 188
  250 CONTINUE                                                        039680  PP-PAM 189
      IUNIT=7                                                         039690  PP-PAM 190
  260 DO 270 I=1,6                                                    039700  PP-PAM 191
      WRITE (IUNIT,450) ((INDEX(I,J,K),K=1,2),J=1,5)                  039710  PP-PAM 192
  270 CONTINUE                                                        039720  PP-PAM 193
      DO 280 I=1,60                                                   039730  PP-PAM 194
      WRITE (IUNIT,460) ((PE(I,J,K),K=1,7),J=1,10)                    039740  PP-PAM 195
  280 CONTINUE                                                        039750  PP-PAM 196
      IF (IUNIT.EQ.7) GO TO 290                                       039760  PP-PAM 197
      IUNIT=7                                                         039770  PP-PAM 198
      GO TO 260                                                       039780  PP-PAM 199
  290 CALL WRITMS (11,ANTX,4760,IECODE,-1,0)                          039790  PP-PAM 200
      CALL CLOSMS (11)                                                039800  PP-PAM 201
      STOP ' IN PAM: NORMAL PROGRAM TERMINATION '                     039810  PP-PAM 202
C                                                                     039820  PP-PAM 203
C                                                                     039830  PP-PAM 204
C   * * *  F O R M A T   S T A T E M E N T S  * * *                   039840  PP-PAM 205
C                                                                     039850  PP-PAM 206
  300 FORMAT (1H ,17HPREPLANNED TARGET)                               039860  PP-PAM 207
  310 FORMAT (1H ,21HTARGET OF OPPORTUNITY)                           039870  PP-PAM 208
  320 FORMAT (F12.8)                                                  039880  PP-PAM 209
  330 FORMAT (1H ,4HETH=,F12.8)                                       039890  PP-PAM 210
  340 FORMAT (1H ,24H TARGET HEADING ANGLE = ,F5.2,13H PT CLS APP= ,F7.2039900  PP-PAM 211
     1)                                                               039910  PP-PAM 212
  350 FORMAT (1H ,6H AZPT=,F5.2,3H V=,F5.2,6H RHO= ,F5.2,5H ED= ,F4.2,5H039920  PP-PAM 213
     1 TR= ,F7.2,6H AOE= ,F5.2)                                       039930  PP-PAM 214
  360 FORMAT (1H ,5H NK= ,I6,6H RNG= ,F10.1,6H ACCX=,F6.2,8H ACCY = ,F6.039940  PP-PAM 215
     11,)                                                             039950  PP-PAM 216
  370 FORMAT (1H ,6HIDRMN=,I4,8H  IDRMX=,I4,7H  IVMX=,I4)             039960  PP-PAM 217

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM - CONT'D

(PAGE 4 OF 11)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

  380 FORMAT (1H ,12HTHETNI(I,J)= /(12F10.1))                              039970  PR-PAM 218
  390 FORMAT (1H0,5H  J= ,I4,9H  NI(J)= ,I4)                               039980  PR-PAM 219
  400 FORMAT (1H ,12H DISMN(I,J)=/(11F10.1))                               039990  PR-PAM 220
  410 FORMAT (1H ,10H IDCODE = ,A10,9H IDCODE= ,O20)                       040000  PR-PAM 221
  420 FORMAT (1H ,10H VELOCITY=,F4.2,16H MEAN RESP TIME=,F7.2,18H DESIGN040010  PR-PAM 222
     1ATOR POWER=,F4.2,15H REFLECTIVITY= ,F4.2)                            040020  PR-PAM 223
  430 FORMAT (1H ,/,1H ,18HDESIGNATION RANGE=,I4,13H  DELTA TIME=,F8.4,2040030  PR-PAM 224
     15H ONE OR SEVERAL  TARGETS =,I4)                                     040040  PR-PAM 225
  440 FORMAT (6F10.5)                                                      040050  PR-PAM 226
  450 FORMAT (2I10)                                                        040060  PR-PAM 227
  460 FORMAT (7F10.5)                                                      040070  PR-PAM 228
      END                                                                  040080  PR-PAM 229

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM - CONT'D

(PAGE 5 OF 11)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE DVDINT (X,FX,XT,FT,NP,ND)                           040100  SR-DVDINT2
      DIMENSION XT(NP), FT(NP), T(16)                                040110  SR-DVDINT3
      N=ND                                                           040120  SR-DVDINT4
      N1=(N-1)/2                                                     040130  SR-DVDINT5
      N2=N/2                                                         040140  SR-DVDINT6
      N3=NP-N2+1                                                     040150  SR-DVDINT7
      IF (NP-N) 250,100,100                                          040160  SR-DVDINT8
  100 N4=N1+2                                                        040170  SR-DVDINT9
      IF (XT(1)-XT(2)) 110,330,260                                   040180  SR-DVDIN10
  110 CONTINUE                                                       040190  SR-DVDIN11
      IF (X-2.*XT(1)+XT(2)) 240,240,120                              040200  SR-DVDIN12
  120 IF (X-2.*XT(NP)+XT(NP-1)) 130,130,240                          040210  SR-DVDIN13
  130 IF (NP.LT.10) GO TO 150                                        040220  SR-DVDIN14
      N5=NP-N                                                        040230  SR-DVDIN15
  140 N5=N5/2                                                        040240  SR-DVDIN16
      N6=N4+N5                                                       040250  SR-DVDIN17
      IF (XT(N6).LT.X) N4=N6                                         040260  SR-DVDIN18
      IF (N5.GT.1) GO TO 140                                         040270  SR-DVDIN19
  150 IF (X-XT(N4)) 180,160,160                                      040280  SR-DVDIN20
  160 IF (N4-N3) 170,180,170                                         040290  SR-DVDIN21
  170 N4=N4+1                                                        040300  SR-DVDIN22
      GO TO 150                                                      040310  SR-DVDIN23
  180 N4=N4-1                                                        040320  SR-DVDIN24
      N5=N4-N1                                                       040330  SR-DVDIN25
      DO 190 I=1,N                                                   040340  SR-DVDIN26
      T(I)=FT(N5)                                                    040350  SR-DVDIN27
  190 N5=N5+1                                                        040360  SR-DVDIN28
      L=(N+1)/2                                                      040370  SR-DVDIN29
      TR=T(L)                                                        040380  SR-DVDIN30
      N6=N4                                                          040390  SR-DVDIN31
      N7=N4+1                                                        040400  SR-DVDIN32
      JU=1                                                           040410  SR-DVDIN33
      N2=N-1                                                         040420  SR-DVDIN34
      UN=1.0                                                         040430  SR-DVDIN35
      DO 230 J=1,N2                                                  040440  SR-DVDIN36
      N5=N4-N1                                                       040450  SR-DVDIN37
      N3=N-J                                                         040460  SR-DVDIN38
      DO 200 I=1,N3                                                  040470  SR-DVDIN39
      N8=N5+J                                                        040480  SR-DVDIN40
      T(I)=(T(I+1)-T(I))/(XT(N8)-XT(N5))                             040490  SR-DVDIN41
  200 N5=N5+1                                                        040500  SR-DVDIN42
      GO TO (210,220), JU                                            040510  SR-DVDIN43
  210 UN=UN*(X-XT(N6))                                               040520  SR-DVDIN44
      JU=2                                                           040530  SR-DVDIN45
      N6=N6-1                                                        040540  SR-DVDIN46
      GO TO 230                                                      040550  SR-DVDIN47
  220 UN=UN*(X-XT(N7))                                               040560  SR-DVDIN48
      JU=1                                                           040570  SR-DVDIN49
      N7=N7+1                                                        040580  SR-DVDIN50
      L=L-1                                                          040590  SR-DVDIN51
  230 TR=TR+UN*T(L)                                                  040600  SR-DVDIN52
      FX=TR                                                          040610  SR-DVDIN53
      RETURN                                                         040620  SR-DVDIN54
  240 WRITE (6,340) X,XT(1),XT(NP)                                   040630  SR-DVDIN55

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM - CONT'D

(PAGE 6 OF 11)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      STOP                                                              040640  SP-DVDIN56
  250 WRITE (6,350) NP,ND                                               040650  SP-DVDIN57
      STOP                                                              040660  SP-DVDIN58
  260 IF (X-2.*XT(1)+XT(2)) 270,240,240                                 040670  SP-DVDIN59
  270 IF (X-2.*XT(NP)+XT(NP-1)) 240,280,280                             040680  SP-DVDIN60
  280 IF (NP.LT.10) GO TO 300                                           040690  SP-DVDIN61
      N5=NP-N                                                           040700  SP-DVDIN62
  290 N5=N5/2                                                           040710  SP-DVDIN63
      N6=N4+N5                                                          040720  SP-DVDIN64
      IF (XT(N6).GT.X) N4=N6                                            040730  SP-DVDIN65
      IF (N5.GT.1) GO TO 290                                            040740  SP-DVDIN66
  300 IF (X-XT(N4)) 310,310,180                                         040750  SP-DVDIN67
  310 IF (N4-N3) 320,180,320                                            040760  SP-DVDIN68
  320 N4=N4+1                                                           040770  SP-DVDIN69
      GO TO 300                                                         040780  SP-DVDIN70
  330 WRITE (6,360) XT(1)                                               040790  SP-DVDIN71
      STOP                                                              040800  SP-DVDIN72
C                                                                       040810  SP-DVDIN73
C   * * *   F O R M A T    S T A T E M E N T S   * * *                  040820  SP-DVDIN74
C                                                                       040830  SP-DVDIN75
  340 FORMAT (23H ARG. NOT IN TABLE  X= ,E14.7,9H  XT(1)= ,E14.7,10H  XT040840  SP-DVDIN76
     1(NP)= ,E14.7,2X,6HDVDINT)                                         040850  SP-DVDIN77
  350 FORMAT (22H TABLE TOO SMALL  NP= ,I5,6H  ND= ,I5,2X,6HDVDINT)     040860  SP-DVDIN78
  360 FORMAT (23H CONSTANT TABLE XT(1)= ,E14.7,2X,6HDVDINT)             040870  SP-DVDIN79
      END                                                               040880  SP-DVDIN80
C                                                                       040890  SP-DVDIN81

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM - CONT'D

(PAGE 7 OF 11)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE NRAN31 (X1,X2,I)                                     040910  SP-NRAN312
      X3=SQRT(-2.0*ALOG(URAN31(I)))                                   040920  SP-NRAN313
      X4=6.2831853072*URAN31(I)                                       040930  SP-NRAN314
      X2=X3*SIN(X4)                                                   040940  SP-NRAN315
      X1=X3*COS(X4)                                                   040950  SP-NRAN316
      RETURN                                                          040960  SP-NRAN317
      END                                                             040970  SP-NRAN318

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM - CONT'D

(PAGE 8 OF 11)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE PENAM2 (TP,IDT,VEL,GTP,PEFL,ANGLET,DEFB,TGTHD,SKSEN,    040990  SR-PENAM22
     1  IDCODE)                                                          041000  SP-PENAM23
C                                                                        041010  SR-PENAM24
      COMMON /XVALUE/ XVALUE(8,9,2)                                      001100  CM+XVALUE2
      DIMENSION AVALUE(9), NP(9)                                         041030  SR-PENAM26
C                                                                        041040  SR-PENAM27
C                                                                        001610  DT+XVALUE2
C     * * * * *   FILL IN XVALUE COMMON BLOCK    * * * * *               001620  DT+XVALUE3
C                                                                        001630  DT+XVALUE4
      DATA (((XVALUE(I,J,K),I=1,8),K=1,2),J=1,9) /                       001640  DT+XVALUE5
C         DUMMY SLOTS FOR NOMINAL RESPONSE TIMES                         001650  DT+XVALUE6
     1    8*0.,                                                          001660  DT+XVALUE7
     2    8*8.,                                                          001670  DT+XVALUE8
C         DESIGNATOR TYPES                                               001680  DT+XVALUE9
     1    1.,     2.,     3.,     5*0.,                                  001690  DT+XVALU10
     2    0.,     1.,     1.,     5*8.,                                  001700  DT+XVALU11
C         TARGET VELOCITIES                                              001710  DT+XVALU12
     1    2.,     3.,     5.,     8.,     9.,     3*0.,                  001720  DT+XVALU13
     2    0.,     0.,     1.,     2.,     2.,     3*8.,                  001730  DT+XVALU14
C         GUN-TARGET RANGES                                              001740  DT+XVALU15
     1    8.,     12.,    20.,    30.,    40.,    10.,    16.,    0.,    001750  DT+XVALU16
     2    0.,     1.,     2.,     3.,     4.,     5.,     6.,     8.,    001760  DT+XVALU17
C         REFLECTIVITIES                                                 001770  DT+XVALU18
     1    .05,    .10,    .20,    .30,    4*0.,                          001780  DT+XVALU19
     2    0.,     1.,     2.,     3.,     4*8.,                          001790  DT+XVALU20
C         ANGLES T                                                       001800  DT+XVALU21
     1    0.,     25.,    30.,    60.,    90.,    120.,   2*0.,          001810  DT+XVALU22
     2    0.,     1.,     2.,     3.,     4.,     5.,     2*8.,          001820  DT+XVALU23
C         DEFLECTION BIASES                                              001830  DT+XVALU24
     1    -200., -100., 0.,       100.,   200.,   3*0.,                  001840  DT+XVALU25
     2    0.,     1.,     2.,     3.,     4.,     3*8.,                  001850  DT+XVALU26
C         TARGET HEADINGS                                                001860  DT+XVALU27
     1    -60.,   -30.,   0.,     30.,    60.,    3*0.,                  001870  DT+XVALU28
     2    0.,     1.,     2.,     3.,     4.,     3*8.,                  001880  DT+XVALU29
C         SEEKER SENSITIVITIES                                           001890  DT+XVALU30
CSEC  THE FOLLOWING CARD CONTAINS FALSE SEEKER SENSITIVITY VALUES TO     001900  DT+XVALU31
CSEC  AVOID INCLUDING CLASSIFIED NUMBERS IN THE PROGRAM LISTING.         001910  DT+XVALU32
CSEC  THE CORRECT VALUES SHOULD BE RESTORED WHEN RUNNING IN A SECURE     001920  DT+XVALU33
CSEC  ENVIRONMENT.  RUNS DONE WITH THESE FALSE VALUES ARE FOR EXAMPLE    001930  DT+XVALU34
CSEC  PURPOSES ONLY.                                                     001940  DT+XVALU35
     1    24.,    36.,    48.,    60.,    72.,    3*0.,                  001950  DT+XVALU36
     2    0.,     1.,     2.,     3.,     4.,     3*8./                  001960  DT+XVALU37
C                                                                        001970  DT+XVALU38
C                                                                        041060  SP-PENAM29
      AVALUE(1)=TP                                                       041070  SP-PENAM10
      AVALUE(2)=IDT                                                      041080  SP-PENAM11
      AVALUE(3)=VEL                                                      041090  SR-PENAM12
      AVALUE(4)=GTR                                                      041100  SP-PENAM13
      AVALUE(5)=REFL                                                     041110  SP-PENAM14
      AVALUE(6)=ANGLET                                                   041120  SP-PENAM15
      AVALUE(7)=DEFB                                                     041130  SR-PENAM16
      AVALUE(8)=TGTHD                                                    041140  SP-PENAM17
      AVALUE(9)=SKSEN                                                    041150  SP-PENAM18
C                                                                        041160  SP-PENAM19

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1  FORTRAN LISTING OF PAM PROGRAM - CONT'D

(PAGE 9 OF 11)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      DO 110 J=2,9                                                      041170  SP-PENAM20
      DO 100 I=1,9                                                      041180  SP-PENAM21
      IF (ABS(AVALUE(J)-XVALUE(I,J,1)).GT..0001) GO TO 100              041190  SP-PENAM22
      NP(J)=XVALUE(I,J,2)+.5                                            041200  SP-PENAM23
      GO TO 110                                                         041210  SP-PENAM24
  100 CONTINUE                                                          041220  SP-PENAM25
      STOP ' IN PENAM2: ERROR NUMBER 1'                                 041230  SP-PENAM26
  110 CONTINUE                                                          041240  SP-PENAM27
C                                                                       041250  SP-PENAM28
      DO 120 J=2,9                                                      041260  SP-PENAM29
      IF (NP(J).GE.8) STOP ' IN PENAM2: ERROR NUMBER 2 '                041270  SP-PENAM30
  120 CONTINUE                                                          041280  SP-PENAM31
C                                                                       041290  SP-PENAM32
      IF (TR.LT.0.0.OR.TR.GE.999.5) STOP ' IN PENAM2: ERROR NUMBER 3 '  041300  SP-PENAM33
      NCODE=TR+.50                                                      041310  SP-PENAM34
C                                                                       041320  SP-PENAM35
      DO 130 J=2,9                                                      041330  SP-PENAM36
  130 NCODE=8*NCODE+NP(J)                                               041340  SP-PENAM37
C                                                                       041350  SP-PENAM38
      ENCODE (10,170,IDCODE) NCODE                                      041360  SP-PENAM39
C                                                                       041370  SP-PENAM40
      RETURN                                                            041380  SP-PENAM41
C                                                                       041390  SP-PENAM42
C  *     *     *     *     *     *     *     *     *     *     *     *  041400  SP-PENAM43
C                                                                       041410  SP-PENAM44
      ENTRY PEIDNT                                                      041420  SP-PENAM45
C                                                                       041430  SP-PENAM46
      DECODE (10,180,IDCODE) NCODE                                      041440  SP-PENAM47
C                                                                       041450  SP-PENAM48
      DO 140 J=2,9                                                      041460  SP-PENAM49
      NP(11-J)=MOD(NCODE,8)                                             041470  SP-PENAM50
      NCODE=NCODE/8                                                     041480  SP-PENAM51
  140 CONTINUE                                                          041490  SP-PENAM52
C                                                                       041500  SP-PENAM53
      NP(1)=NCODE                                                       041510  SP-PENAM54
C                                                                       041520  SP-PENAM55
      DO 160 J=2,9                                                      041530  SP-PENAM56
      DO 150 I=1,8                                                      041540  SP-PENAM57
      IF (ABS(FLOAT(NP(J))-XVALUE(1,J,2)).GT..0001) GO TO 150           041550  SP-PENAM58
      AVALUE(J)=XVALUE(I,J,1)                                           041560  SP-PENAM59
      GO TO 160                                                         041570  SP-PENAM60
  150 CONTINUE                                                          041580  SP-PENAM61
      STOP ' IN PENAM2 (PEIDNT): ERROR NUMBER 4'                        041590  SP-PENAM62
  160 CONTINUE                                                          041600  SP-PENAM63
C                                                                       041610  SP-PENAM64
      AVALUE(1)=NCODE                                                   041620  SP-PENAM65
C                                                                       041630  SP-PENAM66
      TR=AVALUE(1)                                                      041640  SP-PENAM67
      IDT=AVALUE(2)                                                     041650  SP-PENAM68
      VEL=AVALUE(3)                                                     041660  SP-PENAM69
      GTR=AVALUE(4)                                                     041670  SP-PENAM70
      PEFL=AVALUE(5)                                                    041680  SP-PENAM71
      ANGLFT=AVALUE(6)                                                  041690  SP-PENAM72
      DFFR=AVALUE(7)                                                    041700  SP-PENAM73

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM - CONT'D

(PAGE 10 OF 11)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      TGTHD=AVALUE(8)                                                      041710  SP-PENAM74
      SKSFN=AVALUE(9)                                                      041720  SP-PENAM75
C                                                                          041730  SP-PENAM76
      RETURN                                                               041740  SP-PENAM77
C                                                                          041750  SR-PENAM78
C                                                                          041760  SP-PENAM79
C  * * *  F O R M A T   S T A T E M E N T S  * * *                         041770  SP-PENAM80
C                                                                          041780  SP-PENAM81
  170 FORMAT (4HPE00,P6)                                                   041790  SR-PENAM82
  180 FORMAT (4X,P6)                                                       041800  SR-PENAM83
      END                                                                  041810  SP-PENAM84

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING A-1 FORTRAN LISTING OF PAM PROGRAM - CONT'D
(PAGE 11 OF 11)

```
1
.000060
20.0
0.0,0.0
25.,5.,.10,.2,106.
100,8000.,100.,200.
1,7,10
8
0.0,20.,30.,45.,60.,90.,135.,180.
2000.,1800.,1600.,1400.,1200.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
2000.,1800.,1600.,1400.,1200.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
1500.,1400.,1300.,1200.,1100.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
1500.,1400.,1300.,1200.,1100.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
1000.,900.,800.,700.,600.,500.,400.,300.
8
0.,20.,30.,45.,60.,90.,135.,180.
1000.,900.,800.,700.,600.,500.,400.,300.
```

LISTING A-2A SAMPLE INPUT SET 1 FOR PAM

```
1
.000060
20.0
0.0,0.0
25.,3.,.10,.2,106.
100,8000.,100.,200.
1,7,10
8
0.0,20.,30.,45.,60.,90.,135.,180.
2000.,1800.,1600.,1400.,1200.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
2000.,1800.,1600.,1400.,1200.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
1500.,1400.,1300.,1200.,1100.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
1500.,1400.,1300.,1200.,1100.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
1000.,900.,800.,700.,600.,500.,400.,300.
8
0.,20.,30.,45.,60.,90.,135.,180.
1000.,900.,800.,700.,600.,500.,400.,300.
```

LISTING A-2B SAMPLE INPUT SET 2 FOR PAM

```
1
.000060
20.0
0.0,0.0
60.,5.,.30,.20,106.
100,8000.,100.,200.
1,7,10
8
0.0,20.,30.,45.,60.,90.,135.,180.
2000.,1800.,1600.,1400.,1200.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
2000.,1800.,1600.,1400.,1200.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
1500.,1400.,1300.,1200.,1100.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
1500.,1400.,1300.,1200.,1100.,1000.,800.,600.
8
0.,20.,30.,45.,60.,90.,135.,180.
1000.,900.,800.,700.,600.,500.,400.,300.
8
0.,20.,30.,45.,60.,90.,135.,180.
1000.,900.,800.,700.,600.,500.,400.,300.
```

LISTING A-2C SAMPLE INPUT SET 3 FOR PAM

```
JOB CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
ACCOUNT CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
COMMENT.  THIS RUNSTREAM IS USED WITH THE FIRST RUN OF THE PAM PROGRAM;
COMMENT.  THAT IS, THE RUN IN WHICH TAPE11 IS FIRST CREATED.
COMMENT.  THE NEXT STATEMENT DECLARES TAPE5 TO BE A FILE OF RECORD
COMMENT.  TYPE Z WITH 80 CHARACTER LINES.  THIS IS REQUIRED SINCE TAPE5
COMMENT.  IS A FILE FROM THE "FRONT END" COMPUTER WHICH USES A DIFFERENT
COMMENT.  DEFAULT FILE TYPE FROM THE "BACK END" COMPUTER ON WHICH THIS
COMMENT.  JOB IS TO BE RUN.
FILE,TAPE5,RT=Z,FL=80,BT=C.
COMMENT.  THE NEXT STATEMENT OBTAINS A COPY OF THE PAM INPUT FROM THE
COMMENT.  "FRONT END" COMPUTER AND PUTS IT IN THE LOCAL FILE TAPE5.
GETPF,TAPE5,PAMSAMPLEINPUT1,ID=SANDMEYER,ST=MFA.
COMMENT.  THE NEXT STATEMENT ATTACHES THE MACHINE LANGUAGE ("OBJECT
COMMENT.  CODE") CREATED BY RUNNING THE PAM SOURCE CODE THROUGH THE
COMMENT.  FORTRAN COMPILER IN A PREVIOUS JOB.
ATTACH,LGO,PAMSAMPLEBINARY,ID=SANDMEYER.
COMMENT.  THE NEXT STATEMENT RESERVES SPACE ON A PERMANENT FILE DEVICE
COMMENT.  FOR THE FILE WHOSE LOCAL NAME IS TAPE11.
REQUEST,TAPE11,*PF.
COMMENT.  THE NEXT STATEMENT LOADS AND EXECUTES THE PAM PROGRAM USING
COMMENT.  TAPE5 AS THE INPUT FILE.
LGO(TAPE5)
COMMENT.  THE NEXT STATEMENT CATALOGS TAPE11 AS A PERMANENT FILE.
CATALOG,TAPE11,SAMPLETAPE11,ID=SANDMEYER.
COMMENT.  END OF JOB.
```

LISTING A-3A  SAMPLE RUNSTREAM I FOR PAM

```
JOB CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
ACCOUNT CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
COMMENT.  THIS RUNSTREAM IS USED WITH RUNS OF THE PAM PROGRAM MADE AFTER
COMMENT.  TAPE11 HAS BEEN CREATED.
COMMENT.  THE NEXT STATEMENT DECLARES TAPE5 TO BE A FILE OF RECORD
COMMENT.  TYPE Z WITH 80 CHARACTER LINES.  THIS IS REQUIRED SINCE TAPE5
COMMENT.  IS A FILE FROM THE "FRONT END" COMPUTER WHICH USES A DIFFERENT
COMMENT.  DEFAULT FILE TYPE FROM THE "BACK END" COMPUTER ON WHICH THIS
COMMENT.  JOB IS TO BE RUN.
FILE,TAPE5,RT=Z,FL=80,BT=C.
COMMENT.  THE NEXT STATEMENT OBTAINS A COPY OF THE PAM INPUT FROM THE
COMMENT.  "FRONT END" COMPUTER AND PUTS IT IN THE LOCAL FILE TAPE5.
GETPF,TAPE5,PAMSAMPLEINPUT2,ID=SANDMEYER,ST=MFA.
COMMENT.  THE NEXT STATEMENT ATTACHES THE MACHINE LANGUAGE ("OBJECT
COMMENT.  CODE") CREATED BY RUNNING THE PAM SOURCE CODE THROUGH THE
COMMENT.  FORTRAN COMPILER IN A PREVIOUS JOB.
ATTACH,LGO,PAMSAMPLEBINARY,ID=SANDMEYER.
COMMENT.  THE NEXT STATEMENT ATTACHES THE ALREADY EXISTING TAPE11 ("CORE
COMMENT.  DATA BASE" FILE) WHICH WILL BE MODIFIED ("EXTENDED") BY THIS
COMMENT.  RUN.  THE MULTI-READ OPTION IS TURNED OFF SO THAT THE FILE CAN
COMMENT.  BE EXTENDED.
ATTACH,TAPE11,SAMPLETAPE11,ID=SANDMEYER,MR=0.
COMMENT.  THE NEXT STATEMENT LOADS AND EXECUTES THE PAM PROGRAM USING
COMMENT.  TAPE5 AS THE INPUT FILE.
LGO(TAPE5)
COMMENT.  THE NEXT STATEMENT EXTENDS TAPE11.  THIS MEANS THAT THE
COMMENT.  CHANGES MADE BY THIS RUN ARE NOW INCORPORATED IN THE PERMANENT
COMMENT.  FILE THAT WAS ATTACHED UNDER LOCAL FILE NAME TAPE11.
EXTEND,TAPE11.
COMMENT.  END OF JOB.
```

LISTING A-3B  SAMPLE RUNSTREAM 2 FOR PAM

```
JOB CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
ACCOUNT CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
COMMENT.  THIS RUNSTREAM IS USED WITH RUNS OF THE PAM PROGRAM MADE AFTER
COMMENT.  TAPE11 HAS BEEN CREATED.
COMMENT.  THE NEXT STATEMENT DECLARES TAPE5 TO BE A FILE OF RECORD
COMMENT.  TYPE Z WITH 80 CHARACTER LINES.  THIS IS REQUIRED SINCE TAPE5
COMMENT.  IS A FILE FROM THE "FRONT END" COMPUTER WHICH USES A DIFFERENT
COMMENT.  DEFAULT FILE TYPE FROM THE "BACK END" COMPUTER ON WHICH THIS
COMMENT.  JOB IS TO BE RUN.
FILE,TAPE5,RT=Z,FL=80,BT=C.
COMMENT.  THE NEXT STATEMENT OBTAINS A COPY OF THE PAM INPUT FROM THE
COMMENT.  "FRONT END" COMPUTER AND PUTS IT IN THE LOCAL FILE TAPE5.
GETPF,TAPE5,PAMSAMPLEINPUT3,ID=SANDMEYER,ST=MFA.
COMMENT.  THE NEXT STATEMENT ATTACHES THE MACHINE LANGUAGE ("OBJECT
COMMENT.  CODE") CREATED BY RUNNING THE PAM SOURCE CODE THROUGH THE
COMMENT.  FORTRAN COMPILER IN A PREVIOUS JOB.
ATTACH,LGO,PAMSAMPLEBINARY,ID=SANDMEYER.
COMMENT.  THE NEXT STATEMENT ATTACHES THE ALREADY EXISTING TAPE11 ("CORE
COMMENT.  DATA BASE" FILE) WHICH WILL BE MODIFIED ("EXTENDED") BY THIS
COMMENT.  RUN.  THE MULTI-READ OPTION IS TURNED OFF SO THAT THE FILE CAN
COMMENT.  BE EXTENDED.
ATTACH,TAPE11,SAMPLETAPE11,ID=SANDMEYER,MR=0.
COMMENT.  THE NEXT STATEMENT LOADS AND EXECUTES THE PAM PROGRAM USING
COMMENT.  TAPE5 AS THE INPUT FILE.
LGO(TAPE5)
COMMENT.  THE NEXT STATEMENT EXTENDS TAPE11.  THIS MEANS THAT THE
COMMENT.  CHANGES MADE BY THIS RUN ARE NOW INCORPORATED IN THE PERMANENT
COMMENT.  FILE THAT WAS ATTACHED UNDER LOCAL FILE NAME TAPE11.
EXTEND,TAPE11.
COMMENT.  END OF JOB.
```

LISTING A-3C  SAMPLE RUNSTREAM 3 FOR PAM

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

PREPLANNED TARGET
ETH=   .00006000
 AZDT=25.00 V= 5.00 RHO=    .10 ED=  .20 TP=   106.00 AOF=    .35
 TARGET HEADING ANGLE  =  0.00 PT CLS APP=      0.00
 NR=     100 RNG=      8000.0 ACCX=100.00 ACCY =  200.0
IDPMN=    1  IDPMX=    7  IVMX=  10

  J=    1  NI(J)=     8
THEMH(I,J)=
      0.0       20.0       30.0       45.0       60.0       90.0      135.0      180.0
 DISMH(I,J)=
   2000.0     1800.0     1600.0     1400.0     1200.0     1000.0      800.0      600.0

  J=    2  NI(J)=     8
THEMH(I,J)=
      0.0       20.0       30.0       45.0       60.0       90.0      135.0      180.0
 DISMH(I,J)=
   2000.0     1800.0     1600.0     1400.0     1200.0     1000.0      800.0      600.0

  J=    3  NI(J)=     8
THEMH(I,J)=
      0.0       20.0       30.0       45.0       60.0       90.0      135.0      180.0
 DISMH(I,J)=
   1500.0     1400.0     1300.0     1200.0     1100.0     1000.0      800.0      600.0

  J=    4  NI(J)=     8
THEMH(I,J)=
      0.0       20.0       30.0       45.0       60.0       90.0      135.0      180.0
 DISMH(I,J)=
   1500.0     1400.0     1300.0     1200.0     1100.0     1000.0      800.0      600.0

  J=    5  NI(J)=     8
THEMH(I,J)=
      0.0       20.0       30.0       45.0       60.0       90.0      135.0      180.0
 DISMH(I,J)=
   1000.0      900.0      800.0      700.0      600.0      500.0      400.0      300.0

  J=    6  NI(J)=     8
THEMH(I,J)=
      0.0       20.0       30.0       45.0       60.0       90.0      135.0      180.0
 DISMH(I,J)=
   1000.0      900.0      800.0      700.0      600.0      500.0      400.0      300.0
 IDCODE = PEOOAJAAJS IDCODE= 20053333015201011223
 VELOCITY=5.00 MEAN RESP TIME= 106.00 DESIGNATOR POWER= .20 REFLECTIVITY=  .10

DESIGNATION RANGE=    1  DELTA TIME=  0.0000 ONE OR SEVERAL TARGETS =    1
    .29000     .29000     .29000     .29000     .29000     .29000
   1.00000    1.00000    1.00000    1.00000     .91000     .91000
   1.00000    1.00000    1.00000    1.00000     .88000     .88000
   1.00000    1.00000    1.00000    1.00000     .89000     .89000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4  SAMPLE OUTPUT FROM PAM
(PAGE 1 OF 11)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

   1.00000   1.00000   1.00000   1.00000    .89000    .89000
    .99000    .99000    .99000    .99000    .89000    .89000
    .99000    .99000    .99000    .99000    .91000    .91000
   1.00000   1.00000   1.00000   1.00000    .88000    .88000
    .99000    .99000    .99000    .99000    .90000    .90000
   1.00000   1.00000   1.00000   1.00000    .92000    .92000

DESIGNATION RANGE=    1  DELTA TIME=  0.0000 ONE OR SEVERAL TARGETS =    2
    .32000    .32000    .32000    .32000    .32000    .32000
    .99000    .99000    .99000    .99000    .93000    .93000
   1.00000   1.00000   1.00000   1.00000    .93000    .93000
    .99000    .99000    .99000    .99000    .92000    .92000
   1.00000   1.00000   1.00000   1.00000    .91000    .91000
   1.00000   1.00000   1.00000   1.00000    .94000    .94000
   1.00000   1.00000   1.00000   1.00000    .91000    .91000
   1.00000   1.00000   1.00000   1.00000    .91000    .91000
    .99000    .99000    .99000    .99000    .91000    .91000
   1.00000   1.00000   1.00000   1.00000    .91000    .91000

DESIGNATION RANGE=    1  DELTA TIME= 30.0000 ONE OR SEVERAL TARGETS =    1
    .40000    .40000    .40000    .40000    .40000    .40000
    .97000    .97000    .97000    .97000    .60000    .60000
    .97000    .97000    .97000    .97000    .71000    .71000
    .96000    .96000    .96000    .96000    .69000    .69000
    .98000    .98000    .98000    .98000    .71000    .71000
    .97000    .97000    .97000    .97000    .72000    .72000
    .95000    .95000    .95000    .95000    .71000    .71000
    .96000    .96000    .96000    .96000    .58000    .58000
    .97000    .97000    .97000    .97000    .70000    .70000
    .98000    .98000    .98000    .98000    .74000    .74000

DESIGNATION RANGE=    1  DELTA TIME= 90.0000 ONE OR SEVERAL TARGETS =    1
    .19000    .19000    .19000    .19000    .19000    .19000
    .69000    .69000    .69000    .69000    .29000    .29000
    .68000    .68000    .68000    .68000    .27000    .27000
    .74000    .74000    .74000    .74000    .36000    .36000
    .68000    .68000    .68000    .68000    .32000    .32000
    .69000    .69000    .69000    .69000    .28000    .28000
    .70000    .70000    .70000    .70000    .28000    .28000
    .71000    .71000    .71000    .71000    .28000    .28000
    .70000    .70000    .70000    .70000    .27000    .27000
    .72000    .72000    .72000    .72000    .25000    .25000

DESIGNATION RANGE=    1  DELTA TIME=150.0000 ONE OR SEVERAL TARGETS =    1
    .06000    .06000    .06000    .06000    .06000    .06000
    .38000    .38000    .38000    .38000    .14000    .14000
    .36000    .36000    .36000    .36000    .15000    .15000
    .40000    .40000    .40000    .40000    .10000    .10000
    .42000    .42000    .42000    .42000    .09000    .09000
    .40000    .40000    .40000    .40000    .12000    .12000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4 SAMPLE OUTPUT FROM PAM
(PAGE 2 OF 11)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

   .34000    .34000    .34000    .34000    .14000    .14000
   .42000    .42000    .42000    .42000    .04000    .04000
   .36000    .36000    .36000    .36000    .07000    .07000
   .35000    .35000    .35000    .35000    .12000    .12000

DESIGNATION RANGE=   1  DELTA TIME=300.0000 ONE OR SEVERAL TARGETS =    1
   .01000    .01000    .01000    .01000    .01000    .01000
   .05000    .05000    .05000    .05000   0.00000   0.00000
   .04000    .04000    .04000    .04000    .02000    .02000
   .01000    .01000    .01000    .01000    .01000    .01000
   .02000    .02000    .02000    .02000   0.00000   0.00000
   .01000    .01000    .01000    .01000   0.00000   0.00000
   .06000    .06000    .06000    .06000    .01000    .01000
   .05000    .05000    .05000    .05000    .01000    .01000
   .06000    .06000    .05000    .05000    .02000    .02000
   .05000    .05000    .05000    .05000   0.00000   0.00000

DESIGNATION RANGE=   2  DELTA TIME=  0.0000 ONE OR SEVERAL TARGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .92000    .92000    .92000    .92000    .89000    .89000
  1.00000   1.00000   1.00000   1.00000    .87000    .87000
  1.00000   1.00000   1.00000   1.00000    .88000    .88000
  1.00000   1.00000   1.00000   1.00000    .90000    .90000
   .99000    .99000    .99000    .99000    .89000    .89000
   .99000    .99000    .99000    .99000    .88000    .88000
   .98000    .98000    .98000    .98000    .93000    .93000
  1.00000   1.00000   1.00000   1.00000    .86000    .86000
  1.00000   1.00000   1.00000   1.00000    .90000    .90000

DESIGNATION RANGE=   2  DELTA TIME=  0.0000 ONE OR SEVERAL TARGETS =    2
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .97000    .97000    .97000    .97000    .89000    .89000
  1.00000   1.00000   1.00000   1.00000    .91000    .91000
  1.00000   1.00000   1.00000   1.00000    .91000    .91000
  1.00000   1.00000   1.00000   1.00000    .93000    .93000
  1.00000   1.00000   1.00000   1.00000    .91000    .91000
  1.00000   1.00000   1.00000   1.00000    .96000    .96000
  1.00000   1.00000   1.00000   1.00000    .94000    .94000
  1.00000   1.00000   1.00000   1.00000    .91000    .91000
  1.00000   1.00000   1.00000   1.00000    .91000    .91000

DESIGNATION RANGE=   2  DELTA TIME= 30.0000 ONE OR SEVERAL TARGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .91000    .91000    .91000    .91000    .77000    .77000
   .96000    .96000    .96000    .96000    .74000    .74000
   .95000    .95000    .95000    .95000    .68000    .68000
   .99000    .99000    .99000    .99000    .65000    .65000
   .95000    .95000    .95000    .95000    .65000    .65000
   .97000    .97000    .97000    .97000    .73000    .73000
   .99000    .99000    .99000    .99000    .80000    .80000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4 SAMPLE OUTPUT FROM PAM
(PAGE 3 OF 11)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

   .93000    .93000    .93000    .93000    .57000    .57000
   .96000    .96000    .96000    .96000    .76000    .76000

DESIGNATION RANGE=    2  DELTA TIME= 90.0000 ONE OP SEVEPAL TAPGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .56000    .56000    .56000    .56000    .20000    .20000
   .65000    .65000    .65000    .65000    .26000    .26000
   .75000    .75000    .75000    .75000    .29000    .29000
   .66000    .66000    .66000    .66000    .32000    .32000
   .78000    .78000    .78000    .78000    .29000    .29000
   .73000    .73000    .73000    .73000    .28000    .28000
   .68000    .68000    .68000    .68000    .32000    .32000
   .74000    .74000    .74000    .74000    .25000    .25000
   .68000    .68000    .68000    .68000    .34000    .34000

DESIGNATION RANGE=    2  DELTA TIME=150.0000 ONE OP SEVEPAL TAPGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .21000    .21000    .21000    .21000    .06000    .06000
   .39000    .39000    .39000    .39000    .10000    .10000
   .38000    .38000    .38000    .38000    .11000    .11000
   .43000    .43000    .43000    .43000    .07000    .07000
   .41000    .41000    .41000    .41000    .11000    .11000
   .38000    .38000    .38000    .38000    .09000    .09000
   .37000    .37000    .37000    .37000    .09000    .09000
   .31000    .31000    .31000    .31000    .12000    .12000
   .31000    .31000    .31000    .31000    .06000    .06000

DESIGNATION RANGE=    2  DELTA TIME=300.0000 ONE OP SEVEPAL TAPGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .04000    .04000    .04000    .04000   0.00000   0.00000
   .05000    .05000    .05000    .05000    .02000    .02000
   .11000    .11000    .11000    .11000    .01000    .01000
   .06000    .06000    .06000    .06000    .02000    .02000
   .08000    .08000    .08000    .08000    .01000    .01000
   .07000    .07000    .07000    .07000    .01000    .01000
   .02000    .02000    .02000    .02000   0.00000   0.00000
   .05000    .05000    .05000    .05000   0.00000   0.00000
   .01000    .01000    .01000    .01000   0.00000   0.00000

DESIGNATION RANGE=    3  DELTA TIME=  0.0000 ONE OP SEVEPAL TAPGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .50000    .50000    .50000    .50000    .50000    .50000
  1.00000   1.00000   1.00000   1.00000    .94000    .94000
  1.00000   1.00000   1.00000   1.00000    .93000    .93000
   .99000    .99000    .99000    .99000    .89000    .89000
   .99000    .99000    .99000    .99000    .82000    .82000
  1.00000   1.00000   1.00000   1.00000    .90000    .90000
  1.00000   1.00000   1.00000   1.00000    .85000    .85000
  1.00000   1.00000   1.00000   1.00000    .87000    .87000
  1.00000   1.00000   1.00000   1.00000    .89000    .89000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4  SAMPLE OUTPUT FROM PAM

(PAGL 4 OF 11)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

DESIGNATION RANGE=   3  DELTA TIME=  0.0000 ONE OF SEVERAL TARGETS =   2
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .47000    .47000    .47000    .47000    .47000    .47000
  1.00000   1.00000   1.00000   1.00000    .91000    .91000
  1.00000   1.00000   1.00000   1.00000    .95000    .95000
  1.00000   1.00000   1.00000   1.00000    .92000    .92000
  1.00000   1.00000   1.00000   1.00000    .94000    .94000
   .99000    .99000    .99000    .99000    .91000    .91000
  1.00000   1.00000   1.00000   1.00000    .90000    .90000
  1.00000   1.00000   1.00000   1.00000    .92000    .92000
  1.00000   1.00000   1.00000   1.00000    .93000    .93000

DESIGNATION RANGE=   3  DELTA TIME= 30.0000 ONE OF SEVERAL TARGETS =   1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .47000    .47000    .47000    .47000    .47000    .47000
   .95000    .95000    .95000    .95000    .67000    .67000
   .97000    .97000    .97000    .97000    .67000    .67000
   .95000    .95000    .95000    .95000    .68000    .68000
   .96000    .96000    .96000    .96000    .71000    .71000
   .93000    .93000    .93000    .93000    .77000    .77000
   .95000    .95000    .95000    .95000    .69000    .69000
   .98000    .98000    .98000    .98000    .67000    .67000
   .98000    .98000    .98000    .98000    .67000    .67000

DESIGNATION RANGE=   3  DELTA TIME= 90.0000 ONE OF SEVERAL TARGETS =   1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .26000    .26000    .26000    .26000    .26000    .26000
   .75000    .75000    .75000    .75000    .32000    .32000
   .73000    .73000    .73000    .73000    .32000    .32000
   .76000    .76000    .76000    .76000    .39000    .39000
   .68000    .68000    .68000    .68000    .34000    .34000
   .65000    .65000    .65000    .65000    .31000    .31000
   .71000    .71000    .71000    .71000    .26000    .26000
   .66000    .66000    .66000    .66000    .24000    .24000
   .64000    .64000    .64000    .64000    .29000    .29000

DESIGNATION RANGE=   3  DELTA TIME=150.0000 ONE OF SEVERAL TARGETS =   1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .07000    .07000    .07000    .07000    .07000    .07000
   .39000    .39000    .39000    .39000    .08000    .08000
   .35000    .35000    .35000    .35000    .10000    .10000
   .41000    .41000    .41000    .41000    .18000    .18000
   .31000    .31000    .31000    .31000    .05000    .05000
   .42000    .42000    .42000    .42000    .10000    .10000
   .38000    .38000    .38000    .38000    .05000    .05000
   .46000    .46000    .46000    .46000    .14000    .14000
   .41000    .41000    .41000    .41000    .16000    .16000

DESIGNATION RANGE=   3  DELTA TIME=300.0000 ONE OF SEVERAL TARGETS =   1

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4 SAMPLE OUTPUT FROM PAM
(PAGE 5 OF 11)

```
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .01000    .01000    .01000    .01000    .01000    .01000
   .03000    .03000    .03000    .03000    .02000    .02000
   .07000    .07000    .07000    .07000    .02000    .02000
   .03000    .03000    .03000    .03000   0.00000   0.00000
   .02000    .02000    .02000    .02000   0.00000   0.00000
   .04000    .04000    .04000    .04000    .01000    .01000
   .03000    .03000    .03000    .03000    .01000    .01000
   .03000    .03000    .03000    .03000   0.00000   0.00000
   .07000    .07000    .07000    .07000    .01000    .01000

DESIGNATION RANGE=    4   DELTA TIME=  0.0000 ONE OF SEVERAL TARGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .89000    .89000    .89000    .89000    .85000    .85000
   .99000    .99000    .99000    .99000    .84000    .84000
   .99000    .99000    .99000    .99000    .94000    .94000
  1.00000   1.00000   1.00000   1.00000    .93000    .93000
   .99000    .99000    .99000    .99000    .84000    .84000
  1.00000   1.00000   1.00000   1.00000    .93000    .93000
   .99000    .99000    .99000    .99000    .84000    .84000
  1.00000   1.00000   1.00000   1.00000    .85000    .85000

DESIGNATION RANGE=    4   DELTA TIME=  0.0000 ONE OF SEVERAL TARGETS =    2
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .010      .01000    .01000    .01000    .01000    .01000
   .9800     .98000    .98000    .98000    .95000    .95000
  1.0000     .00000   1.00000   1.00000    .93000    .93000
  1.00000    00000    1.00000   1.00000    .94000    .94000
  1.00000    00000    1.00000   1.00000    .93000    .93000
  1.00000    00000    1.00000   1.00000    .93000    .93000
  1.00000    00000    1.00000   1.00000    .88000    .88000
  1.00000    .00000   1.00000   1.00000    .88000    .88000
  1.00000   1.00000   1.00000   1.00000    .95000    .95000

DESIGNATION RANGE=    4   DELTA TIME= 30.0000 ONE OF SEVERAL TARGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .01000    .01000    .01000    .01000    .01000    .01000
   .88000    .88000    .88000    .88000    .77000    .77000
   .98000    .98000    .98000    .98000    .69000    .69000
   .95000    .95000    .95000    .95000    .74000    .74000
   .99000    .99000    .99000    .99000    .62000    .62000
   .99000    .99000    .99000    .99000    .75000    .75000
   .95000    .95000    .95000    .95000    .63000    .63000
   .97000    .97000    .97000    .97000    .65000    .65000
   .96000    .96000    .96000    .96000    .67000    .67000

DESIGNATION RANGE=    4   DELTA TIME= 60.0000 ONE OF SEVERAL TARGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000

0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4 SAMPLE OUTPUT FROM PAM
(PAGE 6 OF 11)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    .57000    .57000    .57000    .57000    .33000    .33000
    .71000    .71000    .71000    .71000    .26000    .26000
    .80000    .80000    .80000    .80000    .36000    .36000
    .74000    .74000    .74000    .74000    .39000    .39000
    .71000    .71000    .71000    .71000    .27000    .27000
    .75000    .75000    .75000    .75000    .32000    .32000
    .73000    .73000    .73000    .73000    .29000    .29000
    .73000    .73000    .73000    .73000    .36000    .36000

DESIGNATION RANGE=    4  DELTA TIME=150.0000 ONE OF SEVERAL TARGETS =    1
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
    .01000    .01000    .01000    .01000    .01000    .01000
    .23000    .23000    .23000    .23000    .10000    .10000
    .40000    .40000    .40000    .40000    .12000    .12000
    .45000    .45000    .45000    .45000    .15000    .15000
    .41000    .41000    .41000    .41000    .13000    .13000
    .37000    .37000    .37000    .37000    .11000    .11000
    .28000    .28000    .28000    .28000    .08000    .08000
    .38000    .38000    .38000    .38000    .12000    .12000
    .35000    .35000    .35000    .35000    .10000    .10000

DESIGNATION RANGE=    4  DELTA TIME=300.0000 ONE OF SEVERAL TARGETS =    1
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
    .01000    .01000    .01000    .01000   0.00000   0.00000
    .06000    .06000    .06000    .06000   0.00000   0.00000
    .05000    .05000    .05000    .05000    .01000    .01000
    .04000    .04000    .04000    .04000    .01000    .01000
    .08000    .08000    .08000    .08000    .01000    .01000
    .04000    .04000    .04000    .04000   0.00000   0.00000
    .05000    .05000    .05000    .05000   0.00000   0.00000
    .02000    .02000    .02000    .02000    .01000    .01000

DESIGNATION RANGE=    5  DELTA TIME=   0.0000 ONE OF SEVERAL TARGETS =    1
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
    .66000    .66000    .66000    .66000    .66000    .66000
    .98000    .98000    .98000    .98000    .87000    .87000
   1.00000   1.00000   1.00000   1.00000    .88000    .88000
    .99000    .99000    .99000    .99000    .90000    .90000
   1.00000   1.00000   1.00000   1.00000    .83000    .83000
    .99000    .99000    .99000    .99000    .94000    .94000
    .99000    .99000    .99000    .99000    .87000    .87000
   1.00000   1.00000   1.00000   1.00000    .88000    .88000

DESIGNATION RANGE=    5  DELTA TIME=   0.0000 ONE OF SEVERAL TARGETS =    2
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
    .69000    .69000    .69000    .69000    .69000    .69000
   1.00000   1.00000   1.00000   1.00000    .91000    .91000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4 SAMPLE OUTPUT FROM PAM
(PAGE 7 OF 11)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

  1.00000   1.00000   1.00000   1.00000    .91000    .91000
  1.00000   1.00000   1.00000   1.00000    .92000    .92000
  1.00000   1.00000   1.00000   1.00000    .95000    .95000
  1.00000   1.00000   1.00000   1.00000    .91000    .91000
  1.00000   1.00000   1.00000   1.00000    .95000    .95000
   .99000    .99000    .99000    .99000    .91000    .91000

DESIGNATION RANGE=   5  DELTA TIME= 30.0000 ONE OF SEVERAL TARGETS =   1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .57000    .57000    .57000    .57000    .57000    .57000
   .96000    .96000    .96000    .96000    .69000    .69000
   .94000    .94000    .94000    .94000    .65000    .65000
   .96000    .96000    .96000    .96000    .70000    .70000
   .95000    .95000    .95000    .95000    .71000    .71000
   .95000    .95000    .95000    .95000    .70000    .70000
   .95000    .95000    .95000    .95000    .73000    .73000
   .97000    .97000    .97000    .97000    .67000    .67000

DESIGNATION RANGE=   5  DELTA TIME= 90.0000 ONE OF SEVERAL TARGETS =   1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .29000    .29000    .29000    .29000    .29000    .29000
   .69000    .69000    .69000    .69000    .22000    .22000
   .72000    .72000    .72000    .72000    .34000    .34000
   .71000    .71000    .71000    .71000    .27000    .27000
   .66000    .66000    .66000    .66000    .23000    .23000
   .77000    .77000    .77000    .77000    .38000    .38000
   .67000    .67000    .67000    .67000    .28000    .28000
   .71000    .71000    .71000    .71000    .29000    .29000

DESIGNATION RANGE=   5  DELTA TIME=150.0000 ONE OF SEVERAL TARGETS =   1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .07000    .07000    .07000    .07000    .07000    .07000
   .36000    .36000    .36000    .36000    .13000    .13000
   .42000    .42000    .42000    .42000    .08000    .08000
   .32000    .32000    .32000    .32000    .14000    .14000
   .34000    .34000    .34000    .34000    .11000    .11000
   .37000    .37000    .37000    .37000    .08000    .08000
   .36000    .36000    .36000    .36000    .06000    .06000
   .46000    .46000    .46000    .46000    .15000    .15000

DESIGNATION RANGE=   5  DELTA TIME=300.0000 ONE OF SEVERAL TARGETS =   1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .04000    .04000    .04000    .04000   0.00000   0.00000
   .05000    .05000    .05000    .05000   0.00000   0.00000
   .06000    .06000    .06000    .06000    .01000    .01000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4 SAMPLE OUTPUT FROM PAM

(PAGE 8 OF 11)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    .03000    .03000    .03000    .03000    .01000    .01000
    .03000    .03000    .03000    .03000    .01000    .01000
    .03000    .03000    .03000    .03000   0.00000   0.00000
    .04000    .04000    .04000    .04000   0.00000   0.00000

DESIGNATION RANGE=   6  DELTA TIME=  0.0000 ONE OF SEVERAL TARGETS =   1
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
    .19000    .19000    .19000    .19000    .19000    .19000
    .92000    .92000    .92000    .92000    .87000    .87000
    .99000    .99000    .99000    .99000    .93000    .93000
   1.00000   1.00000   1.00000   1.00000    .90000    .90000
   1.00000   1.00000   1.00000   1.00000    .93000    .93000
   1.00000   1.00000   1.00000   1.00000    .88000    .88000
   1.00000   1.00000   1.00000   1.00000    .85000    .85000
    .99000    .99000    .99000    .99000    .87000    .87000

DESIGNATION RANGE=   6  DELTA TIME=  0.0000 ONE OF SEVERAL TARGETS =   2
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
    .24000    .24000    .24000    .24000    .24000    .24000
    .97000    .97000    .97000    .97000    .90000    .90000
    .97000    .97000    .97000    .97000    .93000    .93000
   1.00000   1.00000   1.00000   1.00000    .84000    .84000
    .99000    .99000    .99000    .99000    .96000    .96000
   1.00000   1.00000   1.00000   1.00000    .95000    .95000
    .98000    .98000    .98000    .98000    .89000    .89000
   1.00000   1.00000   1.00000   1.00000    .97000    .97000

DESIGNATION RANGE=   6  DELTA TIME= 30.0000 ONE OF SEVERAL TARGETS =   1
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
    .28000    .28000    .28000    .28000    .28000    .28000
    .88000    .88000    .88000    .88000    .77000    .77000
    .98000    .98000    .98000    .98000    .70000    .70000
    .97000    .97000    .97000    .97000    .82000    .82000
    .96000    .96000    .96000    .96000    .73000    .73000
    .94000    .94000    .94000    .94000    .63000    .63000
    .92000    .92000    .92000    .92000    .67000    .67000
    .96000    .96000    .96000    .96000    .77000    .77000

DESIGNATION RANGE=   6  DELTA TIME= 90.0000 ONE OF SEVERAL TARGETS =   1
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
    .16000    .16000    .16000    .16000    .16000    .16000
    .54000    .54000    .54000    .54000    .29000    .29000
    .67000    .67000    .67000    .67000    .28000    .28000
    .73000    .73000    .73000    .73000    .33000    .33000
    .71000    .71000    .71000    .71000    .25000    .25000
    .75000    .75000    .75000    .75000    .29000    .29000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4 SAMPLE OUTPUT FROM PAM

(PAGE 9 OF 11)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

   .73000     .73000     .73000     .73000     .31000     .31000
   .70000     .70000     .70000     .70000     .33000     .33000

DESIGNATION RANGE=   6  DELTA TIME=150.0000 ONE OP SEVERAL TARGETS =   1
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
   .02000     .02000     .02000     .02000     .02000     .02000
   .31000     .31000     .31000     .31000     .11000     .11000
   .41000     .41000     .41000     .41000     .15000     .15000
   .31000     .31000     .31000     .31000     .07000     .07000
   .29000     .29000     .29000     .29000     .08000     .08000
   .43000     .43000     .43000     .43000     .12000     .12000
   .38000     .38000     .38000     .38000     .10000     .10000
   .35000     .35000     .35000     .35000     .11000     .11000

DESIGNATION RANGE=   6  DELTA TIME=300.0000 ONE OP SEVERAL TARGETS =   1
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
   .02000     .02000     .02000     .02000    0.00000    0.00000
   .05000     .05000     .05000     .05000     .01000     .01000
   .05000     .05000     .05000     .05000     .02000     .02000
   .05000     .05000     .05000     .05000     .01000     .01000
   .05000     .05000     .05000     .05000     .02000     .02000
   .05000     .05000     .05000     .05000     .01000     .01000
   .08000     .08000     .08000     .08000     .01000     .01000

DESIGNATION RANGE=   7  DELTA TIME=  0.0000 ONE OP SEVERAL TARGETS =   1
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
   .70000     .70000     .70000     .70000     .70000     .70000
  1.00000    1.00000    1.00000    1.00000     .87000     .87000
  1.00000    1.00000    1.00000    1.00000     .90000     .90000
   .99000     .99000     .99000     .99000     .84000     .84000
  1.00000    1.00000    1.00000    1.00000     .91000     .91000
  1.00000    1.00000    1.00000    1.00000     .91000     .91000
  1.00000    1.00000    1.00000    1.00000     .92000     .92000

DESIGNATION RANGE=   7  DELTA TIME=  0.0000 ONE OP SEVERAL TARGETS =   2
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
  0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
   .80000     .80000     .80000     .80000     .80000     .80000
  1.00000    1.00000    1.00000    1.00000     .93000     .93000
  1.00000    1.00000    1.00000    1.00000     .90000     .90000
   .98000     .98000     .98000     .98000     .91000     .91000
  1.00000    1.00000    1.00000    1.00000     .94000     .94000
  1.00000    1.00000    1.00000    1.00000     .96000     .96000
   .99000     .99000     .99000     .99000     .94000     .94000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4 SAMPLE OUTPUT FROM PAM

(PAGE 10 OF 11)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

DESIGNATION RANGE=    7  DELTA TIME= 30.0000 ONE OF SEVERAL TARGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .62000    .62000    .62000    .62000    .62000    .62000
   .97000    .97000    .97000    .97000    .78000    .78000
   .96000    .96000    .96000    .96000    .72000    .72000
   .98000    .98000    .98000    .98000    .68000    .68000
   .96000    .96000    .96000    .96000    .66000    .66000
   .94000    .94000    .94000    .94000    .67000    .67000
   .97000    .97000    .97000    .97000    .68000    .68000

DESIGNATION RANGE=    7  DELTA TIME= 90.0000 ONE OF SEVERAL TARGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .28000    .28000    .28000    .28000    .28000    .28000
   .71000    .71000    .71000    .71000    .27000    .27000
   .76000    .76000    .76000    .76000    .34000    .34000
   .78000    .78000    .78000    .78000    .34000    .34000
   .70000    .70000    .70000    .70000    .24000    .24000
   .75000    .75000    .75000    .75000    .22000    .22000
   .75000    .75000    .75000    .75000    .30000    .30000

DESIGNATION RANGE=    7  DELTA TIME=150.0000 ONE OF SEVERAL TARGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .11000    .11000    .11000    .11000    .11000    .11000
   .30000    .30000    .30000    .30000    .07000    .07000
   .45000    .45000    .45000    .45000    .08000    .08000
   .32000    .32000    .32000    .32000    .07000    .07000
   .40000    .40000    .40000    .40000    .11000    .11000
   .28000    .28000    .28000    .28000    .07000    .07000
   .27000    .27000    .27000    .27000    .06000    .06000

DESIGNATION RANGE=    7  DELTA TIME=300.0000 ONE OF SEVERAL TARGETS =    1
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
  0.00000   0.00000   0.00000   0.00000   0.00000   0.00000
   .05000    .05000    .05000    .05000    .01000    .01000
   .04000    .04000    .04000    .04000   0.00000   0.00000
   .08000    .08000    .08000    .08000    .01000    .01000
   .06000    .06000    .06000    .06000    .01000    .01000
   .06000    .06000    .06000    .06000    .01000    .01000
   .05000    .05000    .05000    .05000   0.00000   0.00000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING A-4 SAMPLE OUTPUT FROM PAM

(PAGE 11 OF 11)

# APPENDIX B

## LISTING AND SAMPLE CASE OF PRBLOS

# APPENDIX B

## LISTING AND SAMPLE CASE OF PRBLOS

This appendix contains:

(1) A listing of the FORTRAN code for the PRBLOS program.

(2) One set of sample inputs for PRBLOS.

(3) One sample runstream for PRBLOS.

(4) Sample output created by running PRBLOS with the sample input.

Note: The column numbers included in some of the listings in this appendix are not a part of the program code, the data, or the output, but are included only for the reader's convenience.

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      PROGRAM PRBLOS (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
      DIMENSION PRIV(20,2), PRODV(20,2), PRIVM(20,2)
      DIMENSION A(6)
C
  100 WRITE (6,290)
C
      READ (5,220) TGTVEL,TC,(A(I),I=1,6)
      IF (EOF(5).NE.0.0) STOP ' END OF INPUT '
      WRITE (6,230) (A(I),I=1,6)
      WRITE (6,240) TGTVEL,TC
      READ (5,280) NIV
      IF (NIV.GT.20) STOP ' NIV>20 EXCEEDS DIMENSIONS OF PRIV ARRAY '
      DO 110 J=1,2
  110 READ (5,270) (PRIV(I,J),I=1,NIV)
      WRITE (6,250) NIV
      DO 120 J=1,2
  120 WRITE (6,300) (PRIV(I,J),I=1,NIV)
C
      READ (5,280) NODV
      IF (NODV.GT.20) STOP ' NODV>20 EXCEEDS DIMENSIONS OF PRODV ARRAY '
     1
      DO 130 J=1,2
  130 READ (5,270) (PRODV(I,J),I=1,NODV)
      WRITE (6,260) NODV
      DO 140 J=1,2
  140 WRITE (6,300) (PRODV(I,J),I=1,NODV)
C
C
      CD=TC*TGTVEL
C
      DO 150 I=1,NIV
      PRIVM(I,1)=PRIV(I,1)
  150 PRIVM(I,2)=PRIV(I,2)-CD
C
      DO 160 I=1,NIV
      IF (PRIVM(I,2).GT.0.0) GO TO 170
  160 CONTINUE
      WRITE (6,310) CD,PRIV(NIV,2)
      GO TO 210
C
  170 IF (I.EQ.1) GO TO 190
      PRIVM(1,1)=PRIV(I-1,1)+(PRIV(I,1)-PRIV(I-1,1))*(CD-PRIV(I-1,2))
     1  /(PRIV(I,2)-PRIV(I-1,2))
      PRIVM(1,2)=0.0
      NIVM=NIV-I+2
      DO 180 K=2,NIVM
      PRIVM(K,1)=PRIVM(I-2+K,1)
  180 PRIVM(K,2)=PRIVM(I-2+K,2)
  190 CONTINUE
      WRITE (6,320) NIVM
      DO 200 J=1,2
  200 WRITE (6,300) (PRIVM(I,J),I=1,NIVM)
C

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING B-1 FORTRAN LISTING OF PRBLOS PROGRAM

(PAGE 1 OF 3)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

C                                                                           037310  PP-PRBL056
      CALL AVEL (PRIV,20,2,NIV,AVEIV)                                       037320  PR-PRBL057
      CALL AVEL (PRODV,20,2,NODV,AVEODV)                                    037330  PP-PRBL058
      CALL AVEL (PRIVM,20,2,NIVM,AVEIVM)                                    037340  PP-PRBL059
C                                                                           037350  PP-PRBL060
      WRITE (6,330) AVEIV,AVEODV,AVEIVM                                     037360  PP-PRBL061
C                                                                           037370  PR-PRBL062
      PRBIVM=AVEIVM/(AVEIV+AVEODV)                                          037380  PR-PRBL063
      WRITE (6,340) PRBIVM                                                  037390  PR-PRBL064
C                                                                           037400  PP-PRBL065
  210 WRITE (6,350)                                                         037410  PP-PRBL066
      GO TO 100                                                             037420  PR-PRBL067
C                                                                           037430  PP-PRBL068
C                                                                           037440  PP-PRBL069
C  * * *   F O R M A T     S T A T E M E N T S   * * *                      037450  PP-PRBL070
C                                                                           037460  PP-PRBL071
  220 FORMAT (2F10.4,6A10)                                                  037470  PR-PRBL072
  230 FORMAT (7H CASE: ,6A10)                                               037480  PR-PRBL073
  240 FORMAT (10H  TGTVEL= ,F10.4,5X,4HCT= ,F10.4)                          037490  PP-PRBL074
  250 FORMAT (18H PRIV ARRAY: NIV= ,I5)                                     037500  PP-PRBL075
  260 FORMAT (20H PRODV ARRAY: NODV= ,I5)                                   037510  PP-PRBL076
  270 FORMAT (8F10.4)                                                       037520  PP-PRBL077
  280 FORMAT (16I5)                                                         037530  PP-PRBL078
  290 FORMAT (1H1)                                                          037540  PP-PRBL079
  300 FORMAT (1H ,8F12.4)                                                   037550  PP-PRBL080
  310 FORMAT (1H ,//,19H CRITICAL DISTANCE=,F10.4,72H WHICH IS GREATER T037560  PP-PRBL081
     1HAN OR EQUAL TO THE LONGEST POSSIBLE IN-VIEW SEGMENT ,F10.4,/,23H;037570  PR-PRBL082
     2 THEREFORE, PRBLOS=0.0)                                               037580  PP-PRBL083
  320 FORMAT (20H PRIVM ARRAY: NIVM= ,I5)                                   037590  PR-PRBL084
  330 FORMAT (1H ,//,32H AVERAGE IN-VIEW SEGMENT LENGTH=,F10.4,/,36H AVE037600  PR-PRBL085
     1RAGE OUT-OF-VIEW SEGMENT LENGTH=,F10.4,/,41H AVERAGE MODIFIED IN-V037610  PP-PRBL086
     2IEW SEGMENT LENGTH=,F10.4)                                            037620  PP-PRBL087
  340 FORMAT (1H ,//,8H PRBLOS=,F10.4)                                      037630  PR-PRBL088
  350 FORMAT (1H ,//,13H END OF CASE ,//)                                   037640  PP-PRBL089
      END                                                                   037650  PP-PRBL090

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING B-1 FORTRAN LISTING OF PRBLOS PROGRAM - CONT'D

(PAGE 2 OF 3)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE AVEL (P,N1,N2,N3,AVE)                                  037670  SP-AVEL  2
C                                                                       037680  SP-AVEL  3
      DIMENSION P(N1,N2)                                                037690  SP-AVEL  4
C                                                                       037700  SP-AVEL  5
      NU=N3-1                                                           037710  SP-AVEL  6
      SUM=0.0                                                           037720  SP-AVEL  7
      DO 100 I=1,NU                                                     037730  SP-AVEL  8
      SUM=SUM+(P(I+1,2)-P(I,2))*(P(I+1,1)+P(I,1))                       037740  SP-AVEL  9
  100 CONTINUE                                                          037750  SP-AVEL 10
C                                                                       037760  SP-AVEL 11
      AVE=P(N3,1)*P(N3,2)-P(1,1)*P(1,2)-.5*SUM                          037770  SP-AVEL 12
      RETURN                                                            037780  SP-AVEL 13
      END                                                               037790  SP-AVEL 14

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING B-1 FORTRAN LISTING OF PRELOS PROGRAM - CONT'D

(PAGE 3 OF 3)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

3.        15.       CLOSE TERRAIN: RANGE BAND FROM ZERO TO 2000 METERS
  11
0.        .1        .2        .3        .4        .5        .6        .7
.8        .9        1.0
25.       30.       40.       50.       70.       120.      140.      180.
280.      500.      1100.
  13
0.        .61       .743      .8        .85       .865      .874      .9
.91       .95       .98       .99       1.0
0.        25.       50.       75.       100.      125.      150.      175.
200.      400.      600.      800.      1000.
3.        15.       CLOSE TERRAIN: RANGE BAND FROM 2000 TO 2500 METERS
  11
0.        .1        .2        .3        .4        .5        .6        .7
.8        .9        1.0
25.       30.       40.       50.       70.       120.      150.      200.
420.      900.      1520.
  13
0.        .61       .743      .8        .85       .865      .874      .9
.91       .95       .98       .99       1.0
0.        25.       50.       75.       100.      125.      150.      175.
200.      400.      600.      800.      1000.
3.        15.       CLOSE TERRAIN: RANGE BAND FROM 2500 TO 9999 METERS
  11
0.        .1        .2        .3        .4        .5        .6        .7
.8        .9        1.0
25.       40.       60.       100.      140.      190.      210.      240.
310.      590.      1600.
  13
0.        .61       .743      .8        .85       .865      .874      .9
.91       .95       .98       .99       1.0
0.        25.       50.       75.       100.      125.      150.      175.
200.      400.      600.      800.      1000.

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING B-2 SAMPLE INPUT FOR PRBLOS

```
JOB CARD.  CHANGE TO CONFORM TO INSTALLATION USAGE.
ACCOUNT CARD.  CHANGE TO CONFORM TO INSTALLATION USAGE.
COMMENT.  THE FOLLOWING CARD ATTACHES THE MACHINE LANGUAGE (OBJECT CODE) FILE
COMMENT.  OF THE PRBLOS PROGRAM.  THIS FILE WAS CREATED BY RUNNING THE SOURCE
COMMENT.  LISTING OF PRBLOS THROUGH THE FORTRAN COMPILER IN A
COMMENT.  PREVIOUS JOB.
ATTACH,LGO,PRBLOSBINARY,ID=SANDMEYER.
COMMENT.  THE NEXT CARD DECLARES TAPE5 TO BE A FILE OF RECORD TYPE Z WITH
COMMENT.  80 CHARACTER RECORDS.  THIS CARD IS NECESSARY BECAUSE TAPE5 IS TO
COMMENT.  BE OBTAINED FROM THE "FRONT END" COMPUTER WHICH USES A DIFFERENT
COMMENT.  DEFAULT RECORD TYPE FOR ITS FILES.
FILE,TAPE5,RT=Z,FL=80,BT=C.
COMMENT.  THE NEXT CARD OBTAINS A COPY OF TAPE5 (PRBLOS INPUT FILE) FROM
COMMENT.  THE "FRONT END" MACHINE.
GETPF,TAPE5,PRBLOSSAMPLEINPUT,ID=SANDMEYER,ST=MFA.
COMMENT.  THE NEXT CARD LOADS AND EXECUTES THE PROGRAM USING TAPE5 AS INPUT.
LGO(TAPE5)
```

LISTING B-3  SAMPLE RUNSTREAM FOR PRBLOS

```
CASE: CLOSE TERRAIN: RANGE BAND FROM ZERO TO 2000 METERS
 TGTVEL=     3.0000      CT=     15.0000
PRIV ARRAY: NIV=     11
      0.0000       .1000       .2000       .3000       .4000       .5000       .6000       .7000
       .8000       .9000      1.0000
     25.0000     30.0000     40.0000     50.0000     70.0000    120.0000    140.0000    180.0000
    280.0000    500.0000   1100.0000
PROOV ARRAY: NOOV=     13
      0.0000       .6100       .7430       .8000       .8500       .8650       .8740       .9000
       .9100       .9500       .9800       .9900      1.0000
      0.0000     25.0000     50.0000     75.0000    100.0000    125.0000    150.0000    175.0000
    200.0000    400.0000    600.0000    800.0000   1000.0000
PRIVM ARRAY: NIVM=      9
       .2500       .3000       .4000       .5000       .6000       .7000       .8000       .9000
      1.0000
      0.0000      5.0000     25.0000     75.0000     95.0000    135.0000    235.0000    455.0000
   1055.0000

AVERAGE IN-VIEW SEGMENT LENGTH=  197.2500
AVERAGE OUT-OF-VIEW SEGMENT LENGTH=   72.5750
AVERAGE MODIFIED IN-VIEW SEGMENT LENGTH=  155.1250

PRBLOS=     .5749

END OF CASE
```

LISTING B-4  SAMPLE OUTPUT FROM PRBLOS
(PAGE 1 OF 3)

```
CASE: CLOSE TERRAIN: RANGE BAND FROM 2000 TO 2500 METERS
 TGTVEL=     3.0000      CT=     15.0000
PRIV ARRAY: NIV=     11
      0.0000      .1000      .2000      .3000      .4000      .5000      .6000      .7000
       .8000      .9000     1.0000
     25.0000    30.0000    40.0000    50.0000    70.0000   120.0000   150.0000   200.0000
    470.0000   900.0000  1520.0000
PRODV ARRAY: NODV=    13
      0.0000      .6100      .7430      .8000      .8500      .8650      .8740      .9000
       .9100      .9500      .9800      .9900     1.0000
      0.0000    25.0000    50.0000    75.0000   100.0000   125.0000   150.0000   175.0000
    200.0000   400.0000   600.0000   800.0000  1000.0000
PRIVM ARRAY: NIVM=     9
       .2500      .3000      .4000      .5000      .6000      .7000      .8000      .9000
      1.0000
      0.0000     5.0000    25.0000    75.0000   105.0000   155.0000   375.0000   855.0000
   1475.0000

AVERAGE IN-VIEW SEGMENT LENGTH=  275.2500
AVERAGE OUT-OF-VIEW SEGMENT LENGTH=    72.5750
AVERAGE MODIFIED IN-VIEW SEGMENT LENGTH=  233.1250

PRBLOS=     .6702

END OF CASE
```

LISTING B-4 SAMPLE OUTPUT FROM PRBLOS
(PAGE 2 OF 3)

```
CASE: CLOSE TERRAIN: RANGE BAND FROM 2500 TO 9999 METERS
 TGTVEL=     3.0000     CT=     15.0000
PRIV ARRAY: NIV=     11
    0.0000      .1000      .2000      .3000      .4000      .5000      .6000      .7000
     .8000      .9000     1.0000
   25.0000    40.0000    60.0000   100.0000   140.0000   190.0000   210.0000   240.0000
  310.0000   590.0000  1600.0000
PROOV ARRAY: NOOV=     13
    0.0000      .6100      .7430      .8000      .8500      .8650      .8740      .9000
     .9100      .9500      .9800      .9900     1.0000
    0.0000    25.0000    50.0000    75.0000   100.0000   125.0000   150.0000   175.0000
  200.0000   400.0000   600.0000   800.0000  1000.0000
PRIVM ARRAY: NIVM=     10
     .1250      .2000      .3000      .4000      .5000      .6000      .7000      .8000
     .9000     1.0000
    0.0000    15.0000    55.0000    95.0000   145.0000   165.0000   195.0000   265.0000
  545.0000  1555.0000


AVERAGE IN-VIEW SEGMENT LENGTH=  269.2500
AVERAGE OUT-OF-VIEW SEGMENT LENGTH=   72.5750
AVERAGE MODIFIED IN-VIEW SEGMENT LENGTH=  225.5625


PRBLOS=     .6599


END OF CASE
```

LISTING B-4 SAMPLE OUTPUT FROM PRBLOS

(PAGE 3 OF 3)

APPENDIX C

LISTING AND SAMPLE CASE OF PREPMS

# APPENDIX C

## LISTING AND SAMPLE CASE OF PREPMS

This appendix contains:

(1) A listing of the FORTRAN code for the PREPMS program. In addition to the routines in this listing, PREPMS also uses the functions IDCHAR and NUMRIC and the subroutine PENAME which are included in the COPE listing in Appendix D.

(2) One set of sample inputs for PREPMS.

(3) One sample runstream for PREPMS.

(4) Sample output created by running PREPMS with the sample input (and with the records created by the three sample cases of PAM already written to TAPE 11).

Note: The column numbers included in some of the listings in this appendix are not a part of the program code, the data, or the output, but are included only for the reader's convenience.

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)

      PROGRAM PREPMS (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,DEBUG=OUTPUT032250   PP-PREPMS2
     1 ,TAPE4,TAPE11)                                                   032260   PP-PREPMS3
C                                                                       032270   PP-PREPMS4
      COMMON /DODF/ DFDOKL(10,3), NDFSP, NDOKIL, DOKILD                 000250   CM+DODF  2
      LOGICAL DOKILD                                                    000260   CM+DODF  3
      COMMON /INDFX1/ INDX11(2001)                                      000470   CM+INDFX12
      COMMON /HIT/ INDEX(6,5,2), PETBL(60,10,7), DLTT(6), PKTBL(10,2),  000410   CM+HIT   2
     1 NRNGTT, RNGTTF(20), TTF(3,20,2), NPNGPS, RNGPST(10), PSTTBL(10,  000420   CM+HIT   3
     2 3)                                                               000430   CM+HIT   4
      COMMON /RANDOC/ TCRIT, NPLOSR, RNGPLS(10), PPPLOS(10)             000620   CM+RANDOC2
      COMMON /RNGLOS/ NPP, NPP, NRNGCL, CPNGD(11,2), SEGLOS(11,11),     000740   CM+RNGLOS2
     1 RNGCLB(11), VELTBL(2,2)                                          000750   CM+RNGLOS3
      COMMON /RSPTIM/ BATRTH(3), XMVADF(2), DUM1, RCSPTM(2,2), XMTTIM(2,000790   CM+RSPTIM2
     1 2), TMFAN(2,2), TSIGMA(2,2), DETTMA(10,2), NDT, TRARPY(3,2),     000800   CM+RSPTIM3
     2 PRSPTM, TRBAR, PRSDT, PFSVT, ALPHA(2,2), BETA(2,2), DETTIM,      000810   CM+RSPTIM4
     3 IDSGTP, RUNC, PCRNSG, PRDOW                                      000820   CM+RSPTIM5
      COMMON /SADARB/ MAX, PACQ(10), PKILL(10,7), SCANP(10), RADIS,     000860   CM+SADARB2
     1 RTAB(10), BERPX(10), BERPY(10), PEPPX(10), PEPPY(10), NTAB,      000870   CM+SADARB3
     2 IHPSYS                                                           000880   CM+SADARB4
      COMMON /SYMBOL/ BLANK, DECPNT, ALFBET(27), ANUMBR(11), SEP(5),    000970   CM+SYMBOL2
     1 DOLLAR                                                           000980   CM+SYMBOL3
      COMMON /WEATHR/ PPCLCG, PRCFLS(2), PRGCFL(11,2), W(6,11,2),       001060   CM+WEATHP2
     1 VIS(11), CLOUD(6), NCC, NVL, PASQL(6), WNDSPD(3,3), HUMID(2),    001070   CM+WEATHR3
     2 WSUM(2), WCFSUM(2), ICC, IVL                                     001080   CM+WEATHR4
C                                                                       032370   PR-PREPM14
C                                                                       001220   DM+M     2
      DIMENSION WDATA(193), ACQDAT(161), PSPDAT(49), DFSDAT(31)         001230   DM+M     3
      DIMENSION PDDATA(22), PEDATA(4260), PSTDAT(41), AINVDA(167)       001240   DM+M     4
C                                                                       001250   DM+M     5
      EQUIVALENCE (WDATA(1),PPCLCG)                                     001260   DM+M     6
      EQUIVALENCE (ACQDAT(1),NRP)                                       001270   DM+M     7
      EQUIVALENCE (PSPDAT(1),BATPTM(1))                                 001280   DM+M     8
      EQUIVALENCE (DFSDAT(1),DFDOKL(1))                                 001290   DM+M     9
      EQUIVALENCE (PDDATA(1),TCRIT)                                     001300   DM+M    10
      EQUIVALENCE (PEDATA(1),INDEX(1))                                  001310   DM+M    11
      EQUIVALENCE (PSTDAT(1),NRNGPS)                                    001320   DM+M    12
      EQUIVALENCE (AINVDA(1),DLTT(1))                                   001330   DM+M    13
C                                                                       001350   DM+N     2
      DIMENSION WPNDAT(143)                                             001360   DM+N     3
C                                                                       001370   DM+N     4
      EQUIVALENCE (MAY,WPNDAT(1))                                       001380   DM+N     5
C                                                                       001390   DM+N     6
C                                                                       001400   DM+N     7
      EXTERNAL NOFEC2                                                   032400   PP-PREPM17
      LOGICAL NUMPIC                                                    032410   PP-PREPM18
C                                                                       032420   PR-PREPM19
      DIMENSION A(80), E(8,10)                                          032430   PP-PREPM20
      DIMENSION ILIST(2000)                                             032440   PR-PREPM21
      DIMENSION CHAPA(10,30), NOPTNM(70)                                032450   PP-PPEPM22
      DIMENSION IFAY(6)                                                 032460   PP-PREPM23
C                                                                       032470   PP-PREPM24
      DATA IPLT1 /193/ , IPLT2 /161/ , IPLT3 /49/ , IPLT4 /31/ , IPLT5 /032480   PP-PREPM25
     122/ , IPLT6 /4260/ , IPLT7 /41/ , IPLT8 /167/ , ICHPPL /300/      032490   PP-PREPM26
      DATA IPLT9 /143/                                                  032500   PP-PREPM27

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PREPMS PROGRAM

(PAGE 1 OF 12)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)

C                                                                           032510   PR-PREPM28
      IRAY(1)=0                                                             032520   PR-PREPM29
      IRAY(5)=LOCF(NORFC2)                                                  032530   PP-PREPM30
      CALL SYSTEMC (104,IRAY)                                               032540   PP-PREPM31
C                                                                           032550   PP-PREPM32
      WRITE (6,710)                                                         032560   PP-PREPM33
C                                                                           032570   PP-PREPM34
      CALL OPENMS (11,INDX11,2001,1)                                        032580   PP-PREPM35
      CALL READMS (11,NOPTIM,70,PHOPTNUMS)                                  032590   PP-PREPM36
C                                                                           032600   PR-PREPM37
  100 WRITE (6,720)                                                         032610   PP-PREPM38
      READ (5,610) (A(I),I=1,80)                                            032620   PR-PREPM39
      IF (EOF(5).EQ.0.0) GO TO 110                                          032630   PP-PREPM40
      CALL CLOSMS (11)                                                      032640   PP-PREPM41
      STOP ' IN PREPMS: NORMAL PROGRAM TERMINATION '                        032650   PP-PREPM42
C                                                                           032660   PP-PREPM43
  110 WRITE (6,640) (A(I),I=1,80)                                           032670   PR-PREPM44
      CALL SEPPC2 (A,80,B,8,10)                                             032680   PP-PREPM45
      IF (B(1,1).EQ.10HOPTIUNWORD.OR.B(1,1).EQ.6HOPTWDS) GO TO 120          032690   PP-PREPM46
      IF (B(1,1).EQ.10HNEWRECORD .OR.B(1,1).EQ.6HNEWREC) GO TO 180          032700   PP-PREPM47
      IF (B(1,1).EQ.10HREPORTRECO.OR.B(1,1).EQ.10HREPRECLIST) GO TO 540     032710   PP-PREPM48
      IF (B(1,1).EQ.10HINTERPRETE) GO TO 560                                032720   PP-PREPM49
      STOP ' IN PREPMS: ERROR NUMBER 1 '                                    032730   PP-PREPM50
C                                                                           032740   PP-PREPM51
  120 DO 150 J=1,30                                                         032750   PP-PREPM52
      DO 130 I=1,2                                                          032760   PP-PREPM53
  130 CHARA(I,J)=0.0                                                        032770   PP-PREPM54
      DO 140 I=3,10                                                         032780   PR-PREPM55
  140 CHARA(I,J)=BLANK                                                      032790   PP-PREPM56
  150 CONTINUE                                                              032800   PR-PREPM57
      CLABEL=B(1,2)                                                         032810   PP-PREPM58
      WRITE (6,680) CLABEL                                                  032820   PR-PREPM59
      IF (.NOT.NUMRIC(B,8,10,3)) STOP ' IN PREPMS: ERROR NUMBER 2 '         032830   PP-PREPM60
      DECODE (10,630,B(1,3) )X                                              032840   PP-PREPM61
      NREAD=X                                                               032850   PP-PREPM62
      IF (.NOT.NUMRIC(B,8,10,4)) STOP ' IN PREPMS: ERROR NUMBER 3 '         032860   PP-PREPM63
      DECODE (10,630,B(1,4) )Y                                              032870   PP-PREPM64
      NOPT1=Y                                                               032880   PR-PREPM65
      WRITE (6,690) NREAD,NOPT1                                             032890   PP-PREPM66
C                                                                           032900   PP-PREPM67
      DO 170 IJ=1,NREAD                                                     032910   PP-PREPM68
C                                                                           032920   PP-PREPM69
      READ (5,610) (A(I),I=1,80)                                            032930   PP-PREPM70
      WRITE (6,640) (A(I),I=1,80)                                           032940   PP-PREPM71
      CALL SEPPC2 (A,80,B,8,10)                                             032950   PP-PREPM72
      IF (.NOT.NUMRIC(B,8,10,1)) STOP ' IN PREPMS: ERROR NUMBER 4 '         032960   PP-PREPM73
      DECODE (10,630,B(1,1) )OPT                                            032970   PP-PREPM74
      IF (.NOT.NUMRIC(B,8,10,2)) STOP ' IN PREPMS: ERROR NUMBER 5 '         032980   PP-PREPM75
      DECODE (10,630,B(1,2) )XLEV                                           032990   PP-PREPM76
      CHARA(1,IJ)=OPT                                                       033000   PP-PREPM77
      CHARA(2,IJ)=XLEV                                                      033010   PP-PREPM78
      DO 160 IK=1,8                                                         033020   PP-PREPM79
      CHARA(2+IK,IJ)=B(IK,3)                                                033030   PR-PREPM80
  160 CONTINUE                                                              033040   PP-PREPM81

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PREPMS PROGRAM - CONT'D

(PAGE 2 OF 12)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

 170 CONTINUE                                                            033050  PP-PREPM82
C                                                                        033060  PP-PREPM83
      NOPTNM(NOPT1)=NREAD                                                033070  PR-PREPM84
      CALL WRITMS (11,CHAPA,ICHPPL,CLABEL,-1)                            033080  PR-PREPM85
      CALL WRITMS (11,NOPTNM,70,RNOPTNUMS,-1)                            033090  PR-PREPM86
      GO TO 100                                                          033100  PR-PREPM87
C                                                                        033110  PR-PREPM88
 180 FLABEL=R(1,2)                                                       033120  PR-PREPM89
      IF (.NOT.NUMRIC(R,8,10,3)) STOP ' IN PREPMS: ERROR NUMBER 6 '      033130  PR-PREPM90
      DECODE (10,630,R(1,3) )ZCASE                                       033140  PP-PREPM91
      ICASE=ZCASE                                                        033150  PP-PREPM92
      WRITE (6,700) FLABEL,ICASE                                         033160  PP-PREPM93
C                                                                        033170  PR-PREPM94
      GO TO (190,240,290,320,350,370,430,460,500), ICASE                 033180  PR-PREPM95
C                                                                        033190  PP-PREPM96
 190 DO 200 I=1,IPLT1                                                    033200  PR-PREPM97
 200 WDATA(I)=0.0                                                        033210  PR-PREPM98
      READ (5,620) NCC,NVL                                               033220  PR-PREPM99
      WRITE (6,650) NCC,NVL                                              033230  PR-PREP100
      READ (5,630) PRCLCG,(PRCFLS(I),I=1,2)                              033240  PR-PREP101
      WRITE (6,660) PRCLCG,(PRCFLS(I),I=1,2)                             033250  PP-PREP102
      DO 210 J=1,2                                                       033260  PP-PREP103
      READ (5,630) (PRGCFL(I,J),I=1,11)                                  033270  PR-PREP104
 210 WRITE (6,660) (PRGCFL(I,J),I=1,11)                                  033280  PR-PREP105
      DO 220 K=1,2                                                       033290  PP-PREP106
      DO 220 J=1,11                                                      033300  PR-PREP107
      READ (5,630) (W(I,J,K),I=1,6)                                      033310  PR-PREP108
 220 WRITE (6,660) (W(I,J,K),I=1,6)                                      033320  PP-PREP109
      READ (5,630) (VIS(I),I=1,NVL)                                      033330  PP-PREP110
      READ (5,630) (CLOUD(I),I=1,6)                                      033340  PP-PREP111
      WRITE (6,660) (VIS(I),I=1,NVL)                                     033350  PR-PREP112
      WRITE (6,660) (CLOUD(I),I=1,NCC)                                   033360  PP-PREP113
      READ (5,630) (PASQL(I),I=1,6)                                      033370  PR-PREP114
      WRITE (6,660) (PASQL(I),I=1,6)                                     033380  PP-PREP115
      DO 230 J=1,3                                                       033390  PR-PREP116
      READ (5,630) (WNDSPD(I,J),I=1,3)                                   033400  PP-PREP117
      WRITE (6,660) (WNDSPD(I,J),I=1,3)                                  033410  PR-PREP118
 230 CONTINUE                                                            033420  PR-PPEP119
      READ (5,630) (HUMID(I),I=1,2)                                      033430  PP-PREP120
      WRITE (6,660) (HUMID(I),I=1,2)                                     033440  PP-PREP121
      CALL WRITMS (11,WDATA,IPLT1,FLABEL,-1)                             033450  PR-PREP122
      GO TO 530                                                          033460  PR-PREP123
C                                                                        033470  PR-PREP124
 240 DO 250 I=1,IPLT2                                                    033480  PR-PREP125
 250 ACQDAT(I)=0.0                                                       033490  PR-PREP126
      READ (5,620) NRP,NPP,NPNGCL                                        033500  PR-PREP127
      WRITE (6,650) NRP,NPP,NPNGCL                                       033510  PR-PREP128
      NPNGCB=NPNGCL+1                                                    033520  PR-PREP129
      DO 260 J=1,2                                                       033530  PP-PREP130
      READ (5,630) (CRNGD(I,J),I=1,NPP)                                  033540  PR-PREP131
 260 WRITE (6,660) (CRNGD(I,J),I=1,NPP)                                  033550  PR-PREP132
      DO 270 J=1,NPNGCB                                                  033560  PP-PREP133
      READ (5,630) (SFCLOS(I,J),I=1,NPP)                                 033570  PP-PREP134
      WRITE (6,660) (SFCLOS(I,J),I=1,NPP)                                033580  PP-PPEP135

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PREPMS PROGRAM - CONT'D

(PAGE 3 OF 12)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 9 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)

  270 CONTINUE                                                        033590   PP-PREP136
      READ (5,630) (RNGCLB(I),I=1,NRNGCL)                             033600   PP-PREP137
      WRITE (6,660) (RNGCLB(I),I=1,NRNGCL)                            033610   PP-PREP138
      DO 280 J=1,2                                                    033620   PP-PREP139
      READ (5,630) (VFLTBL(I,J),I=1,2)                                033630   PP-PREP140
      WRITE (6,660) (VFLTBL(I,J),I=1,2)                               033640   PP-PREP141
  280 CONTINUE                                                        033650   PP-PREP142
      CALL WRITMS (11,ACQDAT,IRLT2,RLABEL,-1)                         033660   PP-PREP143
      GO TO 530                                                       033670   PP-PREP144
C                                                                     033680   PP-PREP145
  290 DO 300 I=1,IRLT3                                                033690   PP-PREP146
  300 RSPDAT(I)=0.0                                                   033700   PP-PREP147
      READ (5,620) NDT                                                033710   PP-PREP148
      WRITE (6,650) NDT                                               033720   PP-PREP149
      DO 310 J=1,2                                                    033730   PP-PREP150
      READ (5,630) (DETTMA(I,J),I=1,NDT)                              033740   PP-PREP151
      WRITE (6,660) (DETTMA(I,J),I=1,NDT)                             033750   PP-PREP152
  310 CONTINUE                                                        033760   PP-PREP153
      READ (5,630) (BATPTM(I),I=1,3),(XMVADF(I),I=1,2)                033770   PP-PREP154
      WRITE (6,660) (BATPTM(I),I=1,3),(XMVADF(I),I=1,2)               033780   PP-PREP155
      READ (5,630) ((RCSPTM(I,J),I=1,2),J=1,2),((XMTTIM(I,J),I=1,2),J=1,033790   PP-PREP156
     1 2)                                                             033800   PP-PREP157
      WRITE (6,660) ((RCSPTM(I,J),I=1,2),J=1,2),((XMTTIM(I,J),I=1,2),J=1033810   PP-PREP158
     1 ,2)                                                            033820   PP-PREP159
      READ (5,630) ((TMEAN(I,J),I=1,2),J=1,2),((TSIGMA(I,J),I=1,2),J=1, 033830   PP-PREP160
     1 2)                                                             033840   PP-PREP161
      WRITE (6,660) ((TMEAN(I,J),I=1,2),J=1,2),((TSIGMA(I,J),I=1,2),J=1,033850   PP-PREP162
     1 2)                                                             033860   PP-PREP163
      READ (5,630) ((TPAPPY(I,J),I=1,3),J=1,2)                        033870   PP-PREP164
      WRITE (6,660) ((TPAPPY(I,J),I=1,3),J=1,2)                       033880   PP-PREP165
      CALL WRITMS (11,RSPDAT,IRLT3,RLABEL,-1)                         033890   PP-PREP166
      GO TO 530                                                       033900   PP-PREP167
C                                                                     033910   PP-PREP168
  320 DO 330 I=1,IRLT4                                                033920   PP-PREP169
  330 DFSDAT(I)=0.0                                                   033930   PP-PREP170
      READ (5,620) NDFSP                                              033940   PP-PREP171
      WRITE (6,650) NDFSP                                             033950   PP-PREP172
      DO 340 J=1,3                                                    033960   PP-PREP173
      READ (5,630) (DFDOKL(I,J),I=1,NDFSP)                            033970   PP-PREP174
      WRITE (6,660) (DFDOKL(I,J),I=1,NDFSP)                           033980   PP-PREP175
  340 CONTINUE                                                        033990   PP-PREP176
      CALL WRITMS (11,DFSDAT,IRLT4,RLABEL,-1)                         034000   PP-PREP177
      GO TO 530                                                       034010   PP-PREP178
C                                                                     034020   PP-PREP179
  350 DO 360 I=1,IRLT5                                                034030   PP-PREP180
  360 RDDATA(I)=0.0                                                   034040   PP-PREP181
      READ (5,620) NPLOSP                                             034050   PP-PREP182
      READ (5,630) TCRIT                                              034060   PP-PREP183
      READ (5,630) (RNGPLS(I),I=1,NPLOSP)                             034070   PP-PREP184
      READ (5,630) (PPBLOS(I),I=1,NPLOSP)                             034080   PP-PREP185
      WRITE (6,650) NPLOSP                                            034090   PP-PREP186
      WRITE (6,660) TCRIT                                             034100   PP-PREP187
      WRITE (6,660) (RNGPLS(I),I=1,NPLOSP)                            034110   PP-PREP188
      WRITE (6,660) (PPBLOS(I),I=1,NPLOSP)                            034120   PP-PREP189

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 9 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PREPMS PROGRAM - CONT'D

(PAGE 4 OF 12)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      CALL WRITMS (11,PODATA,IPLT5,PLABEL,-1)                               034130  PR-PREP190
      GO TO 530                                                             034140  PP-PREP191
C                                                                           034150  PR-PREP192
  370 DO 380 I=1,IRLT6                                                      034160  PR-PREP193
  380 PFDATA(I)=0.0                                                         034170  PR-PREP194
      IUNIT=5                                                               034180  PR-PREP195
      READ (5,610) (A(I),I=1,80)                                            034190  PR-PREP196
      CALL SEPRC2 (A,80,B,8,10)                                             034200  PR-PREP197
      IF (B(1,1).NE.BHUSETAPE4) GO TO 390                                   034210  PR-PREP198
      IUNIT=4                                                               034220  PR-PREP199
      WRITE (6,600)                                                         034230  PR-PREP200
      GO TO 400                                                             034240  PP-PREP201
  390 ENCODE (20,610,B(1,1) )(A(I),I=1,20)                                  034250  PP-PREP202
      DECODE (20,670,B(1,1) )(INDEX(1,1,K),K=1,2)                           034260  PP-PREP203
  400 CONTINUE                                                              034270  PP-PREP204
      DO 410 I=1,6                                                          034280  PP-PREP205
      ISTART=1                                                              034290  PP-PREP206
      IF (I.EQ.1.AND.IUNIT.EQ.5) ISTART=2                                   034300  PP-PREP207
      DO 410 J=ISTART,5                                                     034310  PP-PREP208
      READ (IUNIT,670) (INDEX(I,J,K),K=1,2)                                 034320  PP-PREP209
      WRITE (6,650) (INDEX(I,J,K),K=1,2)                                    034330  PP-PREP210
  410 CONTINUE                                                              034340  PR-PPEP211
      DO 420 I=1,60                                                         034350  PR-PREP212
      DO 420 J=1,10                                                         034360  PR-PREP213
      READ (IUNIT,630) (PETBL(I,J,K),K=1,7)                                 034370  PR-PREP214
      WRITE (6,660) (PETBL(I,J,K),K=1,7)                                    034380  PR-PREP215
  420 CONTINUE                                                              034390  PR-PPEP216
      GO TO 530                                                             034400  PR-PREP217
C                                                                           034410  PR-PREP218
  430 DO 440 I=1,IRLT7                                                      034420  PR-PREP219
  440 PSTDAT(I)=0.0                                                         034430  PR-PREP220
      READ (5,620) NRNGPS                                                   034440  PR-PREP221
      WRITE ( ,650) NRNGPS                                                  034450  PR-PREP222
      READ (5  ) (RNGPST(I),I=1,NRNGPS)                                     034460  PP-PREP223
      WRITE (   ) (RNGPST(I),I=1,NRNGPS)                                    034470  PR-PREP224
      DO 450 J=  ,                                                          034480  PR-PREP225
      READ (5,6 0) (PSTTBL(I,J),I=1,NRNGPS)                                 034490  PP-PREP226
      WRITE (6,6  TTBL(I,J),I=1,NRNGPS)                                     034500  PP-PREP227
  450 CONTINUE                                                              034510  PR-PREP228
      CALL WRI   STDAT,IPLT7,PLABEL,-1)                                     034520  PR-PREP229
      GO TO 5                                                               034530  PP-PREP230
C                                                                           034540  PP-PREP231
  460 DO 470 I=1,IPLT8                                                      034550  PR-PREP232
  470 AINVDA(I)=0.0                                                         034560  PR-PREP233
      READ (5,620) NRNGTT                                                   034570  PP-PREP234
      WRITE (6,650) NRNGTT                                                  034580  PP-PREP235
      DO 480 J=1,NRNGTT                                                     034590  PP-PREP236
      READ (5,630) RNGTTF(J),((TTF(I,J,K),K=1,2),I=1,3)                     034600  PP-PREP237
  480 WRITE (6,660) RNGTTF(J),((TTF(I,J,K),K=1,2),I=1,3)                    034610  PP-PREP238
      READ (5,630) (DLTT(I),I=1,6)                                          034620  PR-PREP239
      WRITE (6,660) (DLTT(I),I=1,6)                                         034630  PR-PPEP240
      DO 490 J=1,2                                                          034640  PP-PREP241
      READ (5,630) (PKTBL(I,J),I=1,7)                                       034650  PP-PREP242
  490 WRITE (6,660) (PKTBL(I,J),I=1,7)                                      034660  PP-PREP243

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PREPMS PROGRAM - CONT'D

(PAGE 5 OF 12)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      CALL WRITMS (11,AIMVDA,IPLT8,PLABEL,-1)                            034670  PP-PREP244
      GO TO 530                                                          034680  PP-PREP245
C                                                                        034690  PP-PREP246
  500 DO 510 I=1,IPLT9                                                   034700  PP-PREP247
  510 WPNDAT(I)=0.0                                                      034710  PP-PREP248
      READ (5,620) MAX                                                   034720  PP-PREP249
      WRITE (6,650) MAX                                                  034730  PR-PREP250
      READ (5,630) (PACQ(I),I=1,MAX)                                     034740  PR-PREP251
      WRITE (6,660) (PACQ(I),I=1,MAX)                                    034750  PP-PREP252
      DO 520 J=1,7                                                       034760  PR-PREP253
      READ (5,630) (PKILL(I,J),I=1,MAX)                                  034770  PR-PREP254
      WRITE (6,660) (PKILL(I,J),I=1,MAX)                                 034780  PR-PREP255
  520 CONTINUE                                                           034790  PP-PREP256
      READ (5,630) (SCANR(I),I=1,MAX)                                    034800  PR-PREP257
      WRITE (6,660) (SCANP(I),I=1,MAX)                                   034810  PR-PREP258
      READ (5,630) RADIS                                                 034820  PP-PREP259
      WRITE (6,660) RADIS                                                034830  PR-PREP260
      READ (5,620) NTAB                                                  034840  PR-PREP261
      WRITE (6,650) NTAB                                                 034850  PP-PREP262
      READ (5,630) (RTAB(I),I=1,NTAB)                                    034860  PR-PREP263
      WRITE (6,660) (RTAB(I),I=1,NTAB)                                   034870  PR-PREP264
      READ (5,630) (BERRX(I),I=1,NTAB)                                   034880  PR-PREP265
      WRITE (6,660) (BERRX(I),I=1,NTAB)                                  034890  PR-PREP266
      READ (5,630) (BERRY(I),I=1,NTAB)                                   034900  PR-PREP267
      WRITE (6,660) (BERRY(I),I=1,NTAB)                                  034910  PR-PREP268
      READ (5,630) (PERRX(I),I=1,NTAB)                                   034920  PR-PREP269
      WRITE (6,660) (PERRX(I),I=1,NTAB)                                  034930  PR-PREP270
      READ (5,630) (PERRY(I),I=1,NTAB)                                   034940  PR-PREP271
      WRITE (6,660) (PERRY(I),I=1,NTAB)                                  034950  PR-PREP272
      CALL WRITMS (11,WPNDAT,IPLT9,PLABEL,-1)                            034960  PR-PREP273
      GO TO 530                                                          034970  PR-PREP274
C                                                                        034980  PR-PREP275
  530 CONTINUE                                                           034990  PP-PREP276
      GO TO 100                                                          035000  PR-PREP277
C                                                                        035010  PP-PREP278
  540 IOUT=1                                                             035020  PP-PREP279
      DO 550 I=2,2000,2                                                  035030  PR-PREP280
      IF (INDX11(I).EQ.0) GO TO 550                                      035040  PR-PREP281
      ILIST(IOUT)=INDX11(I)                                              035050  PP-PREP282
      ILIST(IOUT+1)=INDX11(I+1)                                          035060  PP-PREP283
      IOUT=IOUT+2                                                        035070  PR-PREP284
  550 CONTINUE                                                           035080  PP-PREP285
C                                                                        035090  PP-PREP286
      WRITE (6,730) (ILIST(IJ),IJ=1,IOUT)                                035100  PP-PREP287
C                                                                        035110  PP-PREP288
      GO TO 100                                                          035120  PP-PREP289
C                                                                        035130  PP-PREP290
  560 CONTINUE                                                           035140  PP-PREP291
      DO 570 I=2,2000,2                                                  035150  PR-PREP292
      DECODE (10,580,INDX11(I) )PECHK                                    035160  PR-PREP293
      IF (PECHK.NE.4HPE00) GO TO 570                                     035170  PP-PREP294
      IPECOD=INDX11(I)                                                   035180  PP-PREP295
      CALL PEIDNT (T1,I2,T3,T4,T5,T6,T7,T8,T9,IPECOD)                    035190  PP-PREP296
      WRITE (6,590) IPECOD,T1,I2,T3,T4,T5,T6,T7,T8,T9                    035200  PP-PREP297

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PREPMS PROGRAM - CONT'D

(PAGE 6 OF 12)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

  570 CONTINUE                                                          035210   PR-PREP298
C                                                                       035220   PR-PREP299
      GO TO 100                                                         035230   PR-PREP300
C                                                                       035240   PR-PREP301
C                                                                       035250   PR-PREP302
C   * * *   F O R M A T    S T A T E M E N T S   * * *                  035260   PR-PREP303
C                                                                       035270   PR-PREP304
  580 FORMAT (A4,6X)                                                    035280   PR-PREP305
  590 FORMAT (5X,11HPE RECORD: ,A10,4H TR=,F4.0,4H DT=,I1,5H VEL=,F2.0,5035290   PR-PREP306
     1H STR=,F3.0,6H REFL=,F4.2,8H ANGLFT=,F4.0,6H DEFB=,F4.0,7H TGTHD=,035300   PR-PREP307
     2F3.0,7H SKSEN=,F3.0)                                              035310   PR-PREP308
  600 FORMAT (1H ,36H TAPE4 USED TO INPUT THIS PE DATA : )              035320   PR-PREP309
  610 FORMAT (80A1)                                                     035330   PR-PREP310
  620 FORMAT (16I5)                                                     035340   PR-PREP311
  630 FORMAT (8F10.4)                                                   035350   PR-PREP312
  640 FORMAT (1H ,80A1)                                                 035360   PR-PREP313
  650 FORMAT (1H ,16I5)                                                 035370   PR-PREP314
  660 FORMAT (1H ,8F10.4)                                               035380   PR-PREP315
  670 FORMAT (2I10)                                                     035390   PR-PREP316
  680 FORMAT (1H ,33H OPTION WORD TYPE RECORD NAMED : ,A10,13H BEING INP035400   PR-PREP317
     1UT )                                                              035410   PR-PREP318
  690 FORMAT (1H ,I2,37H OPTION CARDS READ FOR KEYWORD NUMBER,I3)       035420   PR-PREP319
  700 FORMAT (1H ,16H RECORD NAMED : ,A10,9H OF TYPE ,I2,16H NOW BEING P035430   PR-PREP320
     1EAD )                                                             035440   PR-PREP321
  710 FORMAT (1H1)                                                      035450   PR-PREP322
  720 FORMAT (1H ,8(10H* * * * * ))                                     035460   PR-PREP323
  730 FORMAT (1H ,47H RECORD NAMES AND FILE INFORMATION FOR TAPE11 :,/,(035470   PR-PREP324
     11H ,3(A10,2X,O20,10X)))                                           035480   PR-PREP325
      END                                                               035490   PR-PREP326

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PREPMS PROGRAM - CONT'D

(PAGE 7 OF 12)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      BLOCK DATA BDATA2                                                          035510   BD-BDATA22
C                                                                                035520   BD-BDATA23
      COMMON /LOGFLG/ FIRSTL, SEQNML, SPCL(70), SHRTEC, FTSILL                   000510   CM+LOGFLG2
      LOGICAL FIRSTL, SEQNML, SPCL, SHRTEC, FTSILL                               000520   CM+LOGFLG3
      COMMON /SYMBOL/ BLANK, DECPNT, ALFBET(27), ANUMBR(11), SEP(5),             000970   CM+SYMBOL2
     1  DOLLAR                                                                   000980   CM+SYMBOL3
      COMMON /XVALUE/ XVALUE(8,9,2)                                              001100   CM+XVALUE2
C                                                                                035560   BD-BDATA27
C     * * * * *   FILL IN LOGFLG COMMON BLOCK    * * * * *                       001420   DT+LOGFLG2
C                                                                                001430   DT+LOGFLG3
      DATA FIRSTL /.TRUE./ , SEQNML /.FALSE./                                    001440   DT+LOGFLG4
      DATA (SPCL(I),I=1,70) /70*.FALSE./                                         001450   DT+LOGFLG5
      DATA SHRTEC /.TRUE./                                                       001460   DT+LOGFLG6
C                                                                                001480   DT+SYMBOL2
C     * * * * *   FILL IN SYMBOL COMMON BLOCK    * * * * *                       001490   DT+SYMBOL3
C                                                                                001500   DT+SYMBOL4
      DATA BLANK /1H / , DECPNT /1H./                                            001510   DT+SYMBOL5
      DATA ALFBET /1HA, 1HB, 1HC, 1HD, 1HE, 1HF, 1HG, 1HH, 1HI, 1HJ, 1HK001520   DT+SYMBOL6
     1, 1HL, 1HM, 1HN, 1HO, 1HP, 1HQ, 1HR, 1HS, 1HT, 1HU, 1HV, 1HW, 1HX,001530   DT+SYMBOL7
     2 1HY, 1HZ, 1H*/                                                            001540   DT+SYMBOL8
      DATA ANUMBR /1H0, 1H1, 1H2, 1H3, 1H4, 1H5, 1H6, 1H7, 1H8, 1H9, 1H-001550   DT+SYMBOL9
     1/                                                                          001560   DT+SYMBO10
      DATA SEP /1H,, 1H/, 1H?, 1H(, 1H)/                                         001570   DT+SYMBO11
      DATA DOLLAP /1H$/                                                          001580   DT+SYMBO12
C                                                                                001590   DT+SYMBO13
C                                                                                001610   DT+XVALUE2
C     * * * * *   FILL IN XVALUE COMMON BLOCK    * * * * *                       001620   DT+XVALUE3
C                                                                                001630   DT+XVALUE4
      DATA (((XVALUE(I,J,K),I=1,8),K=1,2),J=1,9) /                               001640   DT+XVALUE5
C          DUMMY SLOTS FOR NOMINAL RESPONSE TIMES                                001650   DT+XVALUE6
     1      8*0.,                                                                001660   DT+XVALUE7
     2      8*8.,                                                                001670   DT+XVALUE8
C          DESIGNATOR TYPES                                                      001680   DT+XVALUE9
     1      1.,     2.,     3.,     5*0.,                                        001690   DT+XVALU10
     2      0.,     1.,     1.,     5*8.,                                        001700   DT+XVALU11
C          TARGET VELOCITIES                                                     001710   DT+XVALU12
     1      2.,     3.,     5.,     8.,     9.,     3*0.,                        001720   DT+XVALU13
     2      0.,     0.,     1.,     2.,     2.,     3*8.,                        001730   DT+XVALU14
C          GUN-TARGET RANGES                                                     001740   DT+XVALU15
     1      8.,     12.,    20.,    30.,    40.,    10.,    16.,    0.,          001750   DT+XVALU16
     2      0.,     1.,     2.,     3.,     4.,     5.,     6.,     8.,          001760   DT+XVALU17
C          REFLECTIVITIES                                                        001770   DT+XVALU18
     1      .05,    .10,    .20,    .30,    4*0.,                                001780   DT+XVALU19
     2      0.,     1.,     2.,     3.,     4*8.,                                001790   DT+XVALU20
C          ANGLES T                                                              001800   DT+XVALU21
     1      0.,     25.,    30.,    60.,    90.,    120.,   2*0.,                001810   DT+XVALU22
     2      0.,     1.,     2.,     3.,     4.,     5.,     2*8.,                001820   DT+XVALU23
C          DEFLECTION BIASES                                                     001830   DT+XVALU24
     1      -200., -100.,  0.,     100.,   200.,   3*0.,                         001840   DT+XVALU25
     2      0.,     1.,     2.,     3.,     4.,     3*8.,                        001850   DT+XVALU26
C          TARGET HEADINGS                                                       001860   DT+XVALU27
     1      -60.,  -30.,   0.,     30.,    60.,    3*0.,                         001870   DT+XVALU28
     2      0.,     1.,     2.,     3.,     4.,     3*8.,                        001880   DT+XVALU29
C          SEEKER SENSITIVITIES                                                  001890   DT+XVALU30

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PRLPMS PROGRAM - CONT'D

(PAGE 8 OF 12)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

CSEC  THE FOLLOWING CARD CONTAINS FALSE SEEKER SENSITIVITY VALUES TO     001900  DT+XVALU31
CSEC  AVOID INCLUDING CLASSIFIED NUMBERS IN THE PROGRAM LISTING.         001910  DT+XVALU32
CSEC  THE CORRECT VALUES SHOULD BE RESTORED WHEN RUNNING IN A SECURE     001920  DT+XVALU33
CSEC  ENVIRONMENT.  RUNS DONE WITH THESE FALSE VALUES ARE FOR EXAMPLE    001930  DT+XVALU34
CSEC  PURPOSES ONLY.                                                     001940  DT+XVALU35
     1      24.,    36.,    48.,    60.,    72.,    3*0.,                001950  DT+XVALU36
     2      0.,     1.,     2.,     3.,     4.,     3*8./                001960  DT+XVALU37
C                                                                        001970  DT+XVALU38
      END                                                                035600  BD-BDATA11

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING C-1  FORTRAN LISTING OF PREPMS PROGRAM - CONT'D

(PAGE 9 OF 12)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE SEPRC2 (A,N1,B,N2,N3)                                  035620   SR-SEPRC22
C                                                                       035630   SP-SEPRC23
      COMMON /LOGFLG/ FIRSTL, SEQNML, SPCL(70), SUPTEC, FTSILL          000510   CM+LOGFLG2
      LOGICAL FIRSTL, SEQNML, SPCL, SUPTEC, FTSILL                      000520   CM+LOGFLG3
      COMMON /SYMBOL/ BLANK, DECPNT, ALFBET(27), ANUMPP(11), SEP(5),    000970   CM+SYMBOL2
     1  DOLLAR                                                          000980   CM+SYMBOL3
C                                                                       035660   SP-SEPRC26
      DIMENSION A(N1), B(N2,N3), NCPP(10)                               035670   SP-SEPRC27
      DIMENSION NC(80), PB(80,10), IC(80,10), NUMFLG(10)                035680   SP-SEPRC28
C                                                                       035690   SP-SEPRC29
      DO 110 I=1,10                                                     035700   SR-SEPRC10
      NUMFLG(I)=0                                                       035710   SR-SEPRC11
      DO 100 J=1,80                                                     035720   SP-SEPRC12
      IC(J,I)=0                                                         035730   SP-SEPRC13
  100 CONTINUE                                                          035740   SP-SEPRC14
  110 CONTINUE                                                          035750   SP-SEPRC15
      DO 120 I=1,80                                                     035760   SP-SEPRC16
  120 NC(I)=0                                                           035770   SP-SEPRC17
      DO 130 I=1,N2                                                     035780   SP-SEPRC18
      DO 130 J=1,N3                                                     035790   SR-SEPRC19
  130 B(I,J)=BLANK                                                      035800   SP-SEPRC20
      DO 140 I=1,80                                                     035810   SP-SEPRC21
      DO 140 J=1,10                                                     035820   SP-SEPRC22
  140 PB(I,J)=BLANK                                                     035830   SP-SEPRC23
      DO 150 I=1,10                                                     035840   SP-SEPRC24
  150 NCPP(I)=0                                                         035850   SR-SEPRC25
C                                                                       035860   SP-SEPRC26
      DO 160 I=1,N1                                                     035870   SP-SEPRC27
  160 NC(I)=IDCHAR(A,N1,I)                                              035880   SR-SEPRC28
C                                                                       035890   SR-SEPRC29
      IF (.NOT.FIRSTL) GO TO 200                                        035900   SP-SEPRC30
      FIRSTL=.FALSE.                                                    035910   SP-SEPRC31
      DO 170 I=73,78                                                    035920   SP-SEPRC32
      IF (NC(I).NE.0) GO TO 230                                         035930   SP-SEPRC33
  170 CONTINUE                                                          035940   SR-SEPRC34
      DO 180 I=63,72                                                    035950   SR-SEPRC35
      IF (NC(I).NE.3) GO TO 230                                         035960   SP-SEPRC36
  180 CONTINUE                                                          035970   SP-SEPRC37
      DO 190 I=79,80                                                    035980   SR-SEPRC38
      IF (NC(I).NE.3) GO TO 230                                         035990   SP-SEPRC39
  190 CONTINUE                                                          036000   SP-SEPRC40
      SEQNML=.TRUE.                                                     036010   SP-SEPRC41
      WRITE (6,330)                                                     036020   SP-SEPRC42
      WRITE (6,340)                                                     036030   SR-SEPRC43
      GO TO 210                                                         036040   SP-SEPRC44
  200 IF (.NOT.SEQNML) GO TO 230                                        036050   SP-SEPRC45
  210 DO 220 I=73,78                                                    036060   SP-SEPRC46
      NC(I)=3                                                           036070   SP-SEPRC47
  220 CONTINUE                                                          036080   SP-SEPRC48
  230 CONTINUE                                                          036090   SP-SEPRC49
C                                                                       036100   SP-SEPRC50
C  SEPARATE RECORDS ON INPUT CARD.                                      036110   SP-SEPRC51
C                                                                       036120   SP-SEPRC52
      I=1                                                               036130   SP-SEPRC53

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PREPMS PROGRAM - CONT'D

(PAGE 10 OF 12)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      IV=0                                                                036140   SP-SEPRC54
      IX=0                                                                036150   SP-SEPRC55
      J=0                                                                 036160   SP-SEPRC56
      LSTCMA=0                                                            036170   SP-SEPRC57
  240 IF (I.GT.N1) GO TO 300                                              036180   SP-SEPRC58
      IF (NC(I).NE.3.OR.IW.NE.IX) GO TO 260                               036190   SP-SEPRC59
  250 I=I+1                                                               036200   SP-SEPRC60
      GO TO 240                                                           036210   SP-SEPRC61
  260 IF (NC(I).EQ.3.OR.NC(I).EQ.4.OR.IW.NE.IX) GO TO 270                 036220   SP-SEPRC62
      IW=IW+1                                                             036230   SP-SEPRC63
      J=0                                                                 036240   SP-SEPRC64
      GO TO 290                                                           036250   SP-SEPRC65
  270 IF (NC(I).NE.4) GO TO 290                                           036260   SP-SEPRC66
      IF (I.NE.(LSTCMA+1)) GO TO 280                                      036270   SP-SEPRC67
      IW=IW+1                                                             036280   SP-SEPRC68
  280 IX=IW                                                               036290   SP-SEPRC69
      LSTCMA=I                                                            036300   SP-SEPRC70
      GO TO 250                                                           036310   SP-SEPRC71
  290 IF (NC(I).EQ.3) GO TO 250                                           036320   SP-SEPRC72
      J=J+1                                                               036330   SP-SEPRC73
      PB(J,IW)=A(I)                                                       036340   SP-SEPRC74
      IC(J,IW)=NC(I)                                                      036350   SP-SEPRC75
      NCPR(IW)=NCPR(IW)+1                                                 036360   SP-SEPRC76
      GO TO 250                                                           036370   SP-SEPRC77
  300 CONTINUE                                                            036380   SP-SEPRC78
      NPF=IW                                                              036390   SP-SEPRC79
      IF (NPE.GE.0.AND.NPE.LE.10) GO TO 310                               036400   SP-SEPRC80
      STOP ' IN SEPRC2: ERROR NUMBER 1 '                                  036410   SP-SEPRC81
C                                                                         036420   SP-SEPRC82
C  ENCODE RECORDS INTO R ARRAY .                                          036430   SP-SEPRC83
C                                                                         036440   SP-SEPRC84
  310 CONTINUE                                                            036450   SP-SEPRC85
      DO 320 I=1,NPE                                                      036460   SP-SEPRC86
      KCHAR=NCPR(I)                                                       036470   SP-SEPRC87
      ENCODE (80,350,R(1,I) )(PB(J,I),J=1,KCHAR)                          036480   SP-SEPRC88
  320 CONTINUE                                                            036490   SP-SEPRC89
C                                                                         036500   SP-SEPRC90
      RETURN                                                              036510   SP-SEPRC91
C                                                                         036520   SP-SEPRC92
C                                                                         036530   SP-SEPRC93
C  * * *  F O R M A T   S T A T E M E N T S  * * *                        036540   SP-SEPRC94
C                                                                         036550   SP-SEPRC95
  330 FORMAT (12H **PREPMS** ,109HNON-FATAL WARNING : YOU HAVE SEQUENCE   036560   SP-SEPRC96
     1NUMBERS IN COLS. 73-78 OF TAPE5.  THEY WILL BE REMOVED BY THE PRO036570   SP-SEPRC97
     2RAM; )                                                              036580   SP-SEPRC98
  340 FORMAT (12H **PREPMS** ,38H HOWEVER, THE RESULTS MAY BE SUSPECT. )036590   SP-SEPRC99
  350 FORMAT (80A1)                                                       036600   SP-SEPR100
      END                                                                 036610   SP-SEPR101

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PRLPMS PROGRAM - CONT'D

(PAGE 11 OF 12)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890.234567890 (READ VERTICALLY)

      SUBROUTINE NORECS                                                  036630  SP-NOREC22
C                                                                        036640  SP-NOREC23
      WRITE (6,100)                                                      036650  SP-NOREC24
C                                                                        036660  SP-NOREC25
      RETURN                                                             036670  SP-NOREC26
C                                                                        036680  SP-NOREC27
C                                                                        036690  SP-NOREC28
C  * * *  F O R M A T   S T A T E M E N T S  * * *                       036700  SP-NOREC29
C                                                                        036710  SP-NOREC10
  100 FORMAT (12H **PREPMS** ,44H THIS IS A CREATION RUN FOR RECORD OPTN036720  SP-NOREC11
     1NUMS )                                                             036730  SP-NOREC12
      END                                                                036740  SP-NOREC13
C                                                                        036750  SP-NOREC14

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789 0 (READ VERTICALLY)
```

LISTING C-1 FORTRAN LISTING OF PREPMS PROGRAM - CONT'D

(PAGE 12 OF 12)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

NEW RECORD,AINVPA0128,8
  16
0.        .99       .95       .99       .80       .90       .60
500.      .95       .90       .95       .75       .85       .55
1000.     .95       .85       .90       .70       .80       .40
1500.     .90       .80       .85       .60       .60       .30
2000.     .85       .80       .85       .65       .75       .30
2500.     .80       .75       .70       .60       .70       .15
3000.     .75       .70       .65       .50       .60       .05
3500.     .70       .65       .60       .45       .0        .0
4000.     .65       .60       .55       .40       .0        .0
4500.     .60       .55       .50       .35       .0        .0
5000.     .55       .50       .45       .30       .0        .0
5500.     .50       .45       .40       .25       .0        .0
6000.     .45       .40       .35       .20       .0        .0
6500.     .40       .35       .30       .15       .0        .0
7000.     .35       .30       .25       .10       .0        .0
9000.     .30       .20       .10       .0        .0        .0
0.        30.       90.       150.      300.      9000.
.500      .700      .500      .800      .700      .950      .600
.500      .600      .450      .750      .700      1.00      .300
NEW RECORD,JUN0600021,1
   6    22
.559      .22       .69
.007      .004      .0        .004      .022      .0        .018      .004
.051      .032      .343
.011      .0        .0        .004      .004      .004      .011      .014
.0        .022      .448
.018      .0        .0        .0        .0        .015
.007      .0        .0        .0        .0        .004
.004      .0        .0        .0        .0        .011
.0        .007      .0        .0        .0        .011
.004      .004      .0        .0        .0        .036
.0        .0        .0        .0        .0        .0
.004      .0        .0        .0        .0        .033
.0        .0        .0        .0        .004      .004
.0        .0        .004      .0        .0        .051
.0        .0        .004      .0        .0        .036
.004      .007      .011      .015      .011      .252
.004      .007      .004      .004      .004      .015
.0        .0        .0        .0        .0        .007
.007      .0        .007      .0        .0        .007
.007      .0        .0        .0        .0        .0
.004      .0        .0        .0        .0        .007
.0        .0        .0        .0        .0        .0
.0        .0        .004      .0        .0        .022
.007      .0        .004      .0        .0        .011
.0        .0        .004      .004      .0        .022
.0        .004      .0        .0        .0        .022

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS
(PAGE 1 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

.011      .007      .004      .004      .018      .212
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
0.        0.        0.        .27       .12       .61
1.        .88       1.
.0        .12       .0
.0        .0        .0
.0        1.
NEW FFC,JUN1400021,1
    6   11
.469      .22       .69
.0        .0        .0        .004      .004      .0        .018      .0
.011      .018      .415
.011      .0        .0        .0        .004      .0        .004      .014
.007      .0        .491
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.004      .0        .0        .0        .0        .0
.0        .0        .0        .0        .004      .0
.0        .0        .0        .0        .0        .0
.004      .004      .004      .0        .0        .007
.0        .0        .0        .0        .0        .0
.0        .0        .004      .007      .0        .0
.0        .0        .0        .0        .004      .014
.007      .004      .022      .014      .117      .256
.007      .0        .0        .0        .0        .004
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.0        .004      .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .004      .0
.007      .004      .0        .0        .004      .0
.0        .0        .0        .004      .004      .0
.0        .0        .0        .0        .0        .0
.004      .004      .04       .036      .213      .195
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
.1'       .22       .26       .37       .0        .0
1.        .39       1.
.0        .36       .0
.0        .25       .0
.63       .37
NEW FFC,JUN2200021,1
    6   11
.52       .22       .69
.01       .0        .0        .0        .017      .0        .028      .017

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D

(PAGE 2 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

.035      .035      .378
.003      .0        .0        .01       .003      .0        .014      .003
.01       .028      .406
.003      .0        .0        .0        .0        .007
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.0        .0        .0        .007      .007      .003
.0        .0        .0        .0        .0        .0
.0        .003      .0        .0        .003      .021
.0        .003      0         .003      .003      .007
.0        .0        .003      .003      .007      .021
.0        .0        .0        .0        .0        .035
.003      .003      .01       .003      .017      .339
.0        .003      .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.003      .0        .0        .0        .003      .003
.003      .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.007      .0        .0        .0        .003      .003
.0        .0        .0        .0        .0        .003
.0        .0        .003      .0        .0        .007
.0        .0        .0        .003      .0        .024
.0        .0        .007      .014      .028      .357
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
0.        0.        0.        .1        .05       .85
1.        1.        1.
.0        .0        .0
.0        .0        .0
.05       .95
NEW PTC,DEC0600021,1
    6    11
.735      .22       .69
.073      .026      .04       .033      .136      .0        .103      .017
.066      .043      .199
.076      .01       .003      .0        .013      .0        .017      .023
.007      .076      .139
.04       .007      .007      .0        .003      .017
.02       .0        .003      .0        .0        .003
.023      .003      .003      .0        .0        .01
.02       .003      .0        .0        .0        .01
.046      .01       .02       .017      .013      .03
.0        .0        .0        .0        .0        .0
.043      .003      .01       .007      .017      .023
.003      .003      .0        .0        .0        .01
.017      .003      .01       .0        .003      .033
.01       .003      .0        .0        .0        .03

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D
(PAGE 3 OF 15)

```
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000001111111111111111111111 COLUMN NOS.
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888899999999990000000000111111111112
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

.007      .0        .007      .017      .02       .149
.01       .003      .003      .003      .003      .003
.01       .0        .0        .0        .0        .0
.003      .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.007      .00       .0        .0        .0        .007
.0        .0        .0        .0        .0        .0
.007      .0        .0        .007      .003      .0
.02       .0        .0        .0        .003      .0
.0        .003      .003      .0        .0        .0
.01       .007      .0        .0        .003      .007
.026      .023      .013      .01       .026      .04
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
.0        .0        .0        .66       .09       .26
1.        .6        1.
.0        .2        .0
.0        .2        .0
.01       .99
NEW PFC,DFC1400021,1
    6   11
.637      .22       .69
.05       .02       .023      .027      .104      .003      .067      .04
.047      .034      .221
.02       .003      .007      .013      .007      .01       .023      .013
.007      .023      .235
.034      .003      .003      .003      .007      .0
.01       .01       .0        .0        .0        .0
.01       .0        .003      .003      .003      .003
.01       .003      .0        .0        .0        .013
.047      .017      .0        .0        .007      .034
.0        .0        .0        .0        .0        .003
.034      .007      .0        .0        .02       .007
.02       .003      .003      .0        .0        .013
.01       .003      .007      .0        .003      .023
.0        .003      .003      .003      .0        .023
.007      .013      .02       .017      .02       .144
.007      .0        .007      .003      .0        .003
.003      .0        .0        .0        .0        .0
.007      .0        .0        .0        .0        .0
.007      .003      .0        .0        .0        .003
.003      .0        .0        .0        .0        .003
.01       .0        .0        .0        .0        .0
.013      .003      .003      .0        .0        .003
.003      .0        .0        .0        .007      .003
.0        .0        .0        .0        .003      .003
.003      .003      .0        .003      .0        .013
.034      .037      .034      .034      .034      .064
1.0       2.        3.        4.        5.        6.        7.        8.

00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000001111111111111111111111 COLUMN NOS.
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888899999999990000000000111111111112
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D
(PAGE 4 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

9.        10.       11.
304.      457.      610.      762.      914.      1372.
.0        .05       .16       .79       .0        .0
1.        .53       1.
.0        .21       .0
.0        .26       .0
.02       .98
NEW REC,DEC2200021,1
     6    11
.757      .22       .69
.083      .027      .037      .033      .123      .0        .107      .013
.063      .04       .230
.023      .0        .007      .007      .017      .0        .027      .013
.017      .013      .12
.033      .0        .003      .007      .003      .037
.023      .0        .003      .0        .0        .0
.02       .0        .01       .0        .0        .007
.013      .0        .003      .0        .003      .013
.05       .02       .007      .01       .02       .017
.0        .0        .0        .0        .0        .0
.03       .017      .007      .0        .017      .037
.003      .0        .0        .0        .007      .003
.01       .01       .007      .003      .007      .027
.003      .0        .0        .003      .007      .027
.007      .01       .027      .023      .03       .133
.01       .003      .0        .01       .0        .0
.0        .0        .0        .0        .0        .0
.007      .0        .0        .0        .0        .0
.007      .0        .0        .0        .0        .0
.007      .0        .003      .003      .0        .003
.0        .0        .0        .0        .0        .0
.007      .003      .0        .0        .003      .013
.01       .0        .003      .0        .0        .0
.007      .0        .0        .0        .0        .01
.0        .003      .007      .0        .0        .003
.017      .013      .01       .01       .01       .06
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
.0        .0        .0        .82       .07       .11
1.        .47       1.
.0        .25       .0
.0        .28       .0
.0        1.
NEW REC,ACQDAT0122,2
    11    11
.0        .1        .2        .3        .4        .5        .6        .7
.8        .9        1.
0.        900.      1500.     2300.     2900.     3400.     3900.     4300.
4600.     4900.     7000.

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D
(PAGE 5 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

.0        .1        .2        .3        .4        .5        .6        .7
.8        .9        1.
25.       40.       70.       90.       150.      230.      300.      420.
580.      1140.     1400.
25.       40.       70.       130.      230.      300.      360.      400.
500.      670.      980.
25.       40.       70.       130.      230.      400.      480.      620.
1000.     1330.     1600.
25.       40.       70.       130.      210.      320.      380.      420.
460.      570.      980.
25.       50.       100.      170.      250.      330.      390.      490.
720.      950.      1425.
1000.     1500.     2000.     2500.     9999.
5.        3.
3.        2.
NEW PFC,ACQDAT0222,2
   11   11    3
.0        .1        .2        .3        .4        .5        .6        .7
.8        .9        1.
.0        400.      650.      900.      1300.     1600.     2000.     2500.
3000.     3700.     7000.
.0        .1        .2        .3        .4        .5        .6        .7
.8        .9        1.
25.       30.       40.       50.       70.       120.      140.      180.
280.      500.      1100.
25.       30.       40.       50.       70.       120.      150.      200.
420.      900.      1520.
25.       40.       60.       100.      140.      190.      210.      240.
310.      590.      1600.
2000.     2500.     9999.
3.        2.
2.        2.
NEW PFC,ACQDAT0322,2
    8   11    1
.0        .02       .04       .1        .2        .3        .32       1.0
0.        2000.     3000.     3500.     5000.     5900.     7000.     9999.
.0        .1        .2        .3        .4        .5        .6        .7
.8        .9        1.
25.       40.       70.       130.      230.      400.      480.      620.
1000.     1330.     1600.
9999.
8.        3.
5.        2.
NEW PFC,FSPDAT0223,3
    9
.0        .125      .25       .375      .5        .625      .75       .875
1.
.0        5.        10.       18.       24.       36.       52.       87.
174.
45.       45.       20.       54.       97.       .975

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D

(PAGE 6 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

17.37     37.37     27.67     47.67     6.        25.       6.        73.
13.57     15.72     32.92     37.13     6.9       6.9       5.95      5.95
 89.      112.       57.       89.      176.       89.
NEW PEC,PSTDAT0127,7
    P
    0.       500.     1000.     1500.     2000.     2500.     3000.     9000.
.07       .07       .13       .15       .16       .17       .18       .18
.9        .9        .82       .78       .76       .74       .73       .73
.03       .03       .05       .07       .08       .09       .09       .09
NEW REC,DFSDAT0124,4
    P
.0        500.      1000.     1500.     2000.     3000.     3001.     9999.
0.        .5        .4        .3        .2        .1        .0        .0
1.        .3        .2        .05       .01       .0        .0        .0
NEW PEC,DFSDAT0224,4
    2
0.        9999.
0.        0.
0.        0.
OPTION WORDS,NEXTCASE,4,2
1,2,RESETDEFAULTS
1,2,RD
2,2,DON'T RESET
2,2,DR
OPT WDS,PLOT,4,5
1,2,PRINTER
2,2,CALCOMP
1,3,OFF
2,3,ON
OPT WDS,LOSMODE,4,11
1,2,SHOOTINGGALLERY
1,2,SG
2,2,RANDOMOCCURRENCE
2,2,RO
OPT WDS,MSNCODE,6,12
1,2,PRIORITYPREPLANNEDTARGET
1,2,PRPT
2,2,PREPLANNEDTARGET
2,2,PPT
3,2,TARGETOFOPPORTUNITY
3,2,TOO
OPTWDS,NUMDIFTGTP,4,13
1,2,THREE
1,2,3
2,2,TWO
2,2,2
OPTWDS,TARGETTYPE,7,14
1,2,1
2,2,2
3,2,3

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2  SAMPLE INPUT FOR PREPMS - CONT'D

(PAGE 7 OF 15)

```
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

4,2,4
5,2,5
6,2,6
7,2,7
OPTWDS,DESIGTYPE,6,16
1,2,GILD
1,2,G
2,2,MULE
2,2,M
3,2,LTD
3,2,L
OPTWDS,FOPOSITION,4,17
1,2,VANTAGEPOINT
1,2,VP
2,2,MANEUVERUNIT
2,2,MU
OPTWDS,TLEFACTORS,1,33
1,1,NOTHING
OPTWDS,ACQRNGDIST,8,22
1,2,AVERAGE
1,2,A
2,2,CLOSE
2,2,C
3,2,OPEN
3,2,O
4,2,STAGS
4,2,S
OPTWDS,RESPTIME,15,23
2,2,STAUCH
2,2,S
1,2,FTSILL
1,2,FS
3,2,SCHNGIO
11,2,PAPAM
11,2,P
1,3,DIGITAL
1,3,D
2,3,VOICE
2,3,V
1,4,DAY
1,4,D
2,4,NIGHT
2,4,N
OPT WDS,DFIRESUPPE,4,24
1,2,HIGH
1,2,H
2,2,NONE
2,2,N
OPTWDS,PUMS,2,31
1,2,NO

0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D
(PAGE 8 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

1,2,H
OPTWDS,CONTROL,3,9
1,2,ECHO
1,3,SHORT
2,3,LONG
OPTWDS,SMOKE,4,32
1,2,SMOKEROUNDSFIRED
1,2,SMKRNDS
2,2,PROBABILITYSMOKEKILLSMISSION
2,2,PFBSMOKE
NEW PFC,PSPOAT0123,3
    9
0.        .125      .25       .375      .5        .625      .75       .875
1.
0.        5.        10.       18.       24.       36.       52.       87.
174.
20.       20.       20.       0.        0.        1.0
0.        0.        0.        0.        0.        0.        0.        0.
69.19     69.19     69.19     69.19     46.32     46.32     46.32     46.32
 75.       75.       75.       75.       75.       75.
NEW PFC,PODAV3TPN1,5
    5
   15.
1000.     1500.     2000.     2500.     9999.
.7392     .7040     .8008     .6962     .7617
NEW PFC,PODAV5TPN1,5
    5
   15.
1000.     1500.     2000.     2500.     9999.
.6831     .6403     .7580     .6309     .7078
NEW PFC,PODAV8TPN1,5
    5
  15.
1000.     1500.     2000.     2500.     9999.
.6129     .5539     .6998     .5422     .6336
NEW PFC,PODAV3TPN2,5
    3
   15.
2000.     2500.     9999.
.5749     .6702     .6599
NEW PFC,PODAV5TPN2,5
    5
   15.
2000.     2500.     9999.
.5023     .6139     .5888
NEW PFC,PODAV8TPN2,5
    3
   15.
2000.     2500.     9999.
.4114     .5434     .4959

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D

(PAGE 9 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

NEW REC,RODAV3TFN3,5
    1
   15.
 9999.
.8008
NEW REC,RODAV5TFN3,5
    1
   15.
 9999.
.7580
NEW REC,RODAV8TFN3,5
    1
   15.
 9999.
.6998
NEW REC,RSPDAT0323,3
    9
0.0       .125      .25       .375      .5        .625      .75       .875
1.0
0.0       5.        10.       18.       24.       36.       52.       87.
174.
0.0       0.0       0.0       0.0       0.0       .975
50.       50.       50.       50.       0.        0.        0.        0.
159.      159.      159.      159.      80.       80.       80.       80.
209.      209.      209.      209.      209.      209.
NEWREC,MAR0600021,1
    6    11
.659      .22       .69
.032      .01       .023      .039      .042      .0        .071      .01
.087      .068      .28
.026      .01       .0        .019      .0        .0        .029      .019
.035      .029      .174
.013      .0        .0        .0        .003      .016
.010      .0        .0        .0        .0        .0
.006      .006      .006      .0        .0        .003
.006      .0        .01       .003      .003      .016
.006      .01       .0        .0        .0        .026
.0        .0        .0        .0        .0        .0
.013      .013      .0        .006      .006      .032
.0        .003      .0        .0        .003      .003
.013      .01       .003      .006      .013      .042
.006      .006      .01       .0        .0        .045
.000      .000      .016      .013      .026      .275
.003      .0        .003      .0        .0        .019
.006      .0        .0        .0        .0        .003
.0        .0        .0        .0        .0        .0
.006      .0        .003      .0        .0        .01
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.006      .0        .0        .0        .0        .023

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D
(PAGE 10 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

.003      .0        .0        .003      .0        .013
.006      .003      .0        .0        .003      .023
.0        .003      .0        .0        .003      .023
.003      .003      .016      .0        .026      .125
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1377.
0.        0.        0.        .65       .11       .24
1.        .51       1.
0.        .30       0.
0.        .19       0.
.009      .991
NEWREC,MAP1400021,1
    6    11
.598      .22       .69
.009      .003      .006      .003      .032      .003      .019      .009
.035      .06       .418
.0        .0        .003      .006      .003      .0        .009      .016
.022      .009      .332
.009      .0        .0        .0        .0        .0
.0        .003      .0        .0        .0        .0
.003      .0        .0        .0        .0        .003
.0        .0        .0        .0        .003      .0
.009      .009      .003      .003      .0        .006
.0        .0        .0        .0        .0        .003
.003      .003      .0        .0        .003      .009
.0        .0        .003      .0        .0        .006
.003      .0        .013      .0        .003      .016
.0        .003      .003      .009      .013      .032
.019      .013      .028      .051      .073      .234
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.003      .0        .0        .0        .0        .0
.003      .0        .003      .0        .0        .0
.0        .0        .0        .0        .003      .0
.0        .0        .0        .0        .0        .0
.006      .0        .0        .0        .003      .0
.0        .0        .0        .003      .0        .013
.0        .009      .006      .003      .0        .003
.0        .003      .0        .0        .0        .006
.006      .013      .025      .06       .108      .120
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1377.
.04       .15       .21       .60       .0        .0
1.        .3        1.
0.        .28       .0
.0        .42       .0
.377      .623
NEWREC,MAP2200021,1

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D
(PAGE 11 OF 15)

AD-A100 285 ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY ABERDEEN PROV--ETC F/G 19/1
COPPERHEAD OPERATIONAL PERFORMANCE EVALUATION (COPE): COMPUTER --ETC(U)
MAR 81 R S SANDMEYER

UNCLASSIFIED AMSAA-TR-318 NL

5 of 6

AD A
100285

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000011111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

.003      .0        .0        .003      .0        .013
.006      .003      .0        .0        .003      .023
.0        .003      .0        .0        .003      .023
.003      .003      .016      .0        .026      .125
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
0.        0.        0.        .65       .11       .24
1.        .51       1.
0.        .30       0.
0.        .19       0.
.009      .991
NEWPFC,MAP1400021,1
    6    11
.598      .22       .69
.009      .003      .006      .003      .032      .003      .019      .009
.035      .06       .418
.0        .0        .003      .006      .003      .0        .009      .016
.022      .009      .332
.009      .0        .0        .0        .0        .0
.0        .003      .0        .0        .0        .0
.003      .0        .0        .0        .0        .003
.0        .0        .0        .0        .003      .0
.009      .009      .003      .003      .0        .006
.0        .0        .0        .0        .0        .003
.003      .003      .0        .0        .003      .009
.0        .0        .003      .0        .0        .006
.003      .0        .013      .0        .003      .016
.0        .003      .003      .009      .013      .032
.019      .013      .028      .051      .073      .234
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.003      .0        .0        .0        .0        .0
.003      .0        .003      .0        .0        .0
.0        .0        .0        .0        .003      .0
.0        .0        .0        .0        .0        .0
.006      .0        .0        .0        .003      .0
.0        .0        .0        .003      .0        .013
.0        .009      .006      .003      .0        .003
.0        .003      .0        .0        .0        .006
.006      .013      .025      .06       .108      .170
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
.04       .15       .21       .60       .0        .0
1.        .3        1.
0.        .28       .0
.0        .42       .0
.377      .623
NEWPFC,MAP2200021,1

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000011111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D
(PAGE 11 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    6   11
.679      .22       .69
.016      .003      .006      .003      .042      .003      .032      .019
.077      .087      .391
.01       .0        .0        .003      .003      .0        .032      .01
.016      .022      .224
.0        .0        .0        .0        .0        .016
.003      .0        .0        .0        .0        .0
.003      .003      .0        .0        .0        .0
.0        .0        .0        .0        .0        .003
.01       .003      .003      .006      .003      .016
.0        .0        .0        .0        .0        .003
.0        .0        .0        .003      .0        .029
.0        .0        .0        .003      .0        .016
.006      .01       .01       .003      .003      .045
.0        .0        .006      .0        .006      .074
.0        .003      .013      .013      .038      .374
.0        .0        .0        .003      .0        .006
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.003      .0        .0        .0        .0        .0
.003      .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.022      .0        .0        .0        .003      .006
.0        .003      .0        .0        .0        .006
.0        .01       .0        .003      .0        .003
.0        .003      .0        .003      .006      .01
.006      .003      .013      .006      .016      .179
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
0.        0.        0.        .58       .08       .34
1.        .42       1.
.0        .27       .0
.0        .31       .0
.0        1.0
NEWPEC,SEP0600021,1
    6   11
.568      .22       .69
.129      .025      .036      .072      .043      .0        .04       .0
.05       .04       .183
.04       .007      .014      .018      .032      .0        .032      .011
.036      .022      .219
.101      .0        .0        .0        .0        .029
.007      .0        .0        .004      .0        .014
.004      .0        .007      .0        .0        .025
.0        .0        .0        .0        .0        .022
.004      .0        .0        .0        .004      .036
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .004      .036

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2  SAMPLE INPUT FOR PREPMS - CONT'D
(PAGE 12 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

.0        .0        .0        .0        .0        .0
.0        .004      .0        .0        .007      .04
.0        .0        .0        .0        .004      .036
.0        .064      .004      .011      .018      .147
.014      .0        .0        .0        .0        .025
.007      .0        .0        .0        .0        .0
.0        .004      .0        .0        .0        .011
.004      .0        .0        .0        .0        .014
.011      .0        .0        .004      .0        .018
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .032
.0        .0        .0        .0        .0        .011
.004      .0        .0        .0        .004      .029
.0        .004      .0        .004      .007      .007
.022      .007      .011      .004      .022      .155
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
.0        .0        .0        .57       .11       .32
1.        .79       1.
.0        .15       .0
.0        .06       .0
.0        1.0
NEWEFC,SEP1400021,1
    6   11
.437      .22       .69
.0        .0        .0        .007      .01       .003      .014      .0
.014      .031      .357
.007      .0        .0        .0        .01       .0        .01       .021
.003      .024      .488
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.003      .003      .0        .0        .0        .0
.003      .0        .0        .0        .0        .007
.0        .0        .0        .0        .0        .003
.0        .0        .0        .0        .03       .01
.0        .0        .0        .0        .0        .0
.0        .0        .0        .003      .003      .007
.0        .0        .003      .003      .0        .074
.003      .003      .007      .007      .045      .202
.003      .0        .0        .0        .0        .003
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.003      .003      .0        .0        .0        .003
.0        .0        .0        .0        .0        .0
.003      .0        .0        .0        .007      .0
.003      .0        .0        .01       .003      .003
.0        .0        .0        .0        .0        .003

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D
(PAGE 13 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

.0        .0        .007      .0        .007      .01
.007      .017      .031      .034      .220      .179
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
.04       .21       .29       .46       .0        .0
1.        .43       1.
.0        .32       .0
.0        .25       .0
.547      .453
NEWPEC,SEP2200021,1
    6    11
.602      .22       .49
.013      .0        .003      .01       .024      .0        .04       .01
.047      .071      .384
.020      .0        .003      .0        .01       .003      .02       .003
.017      .04       .279
.0        .0        .0        .0        .0        .013
.0        .0        .0        .0        .0        .0
.0        .0        .003      .0        .0        .0
.0        .0        .0        .0        .0        .01
.003      .0        .0        .0        .0        .02
.0        .0        .0        .0        .0        .0
.003      .0        .0        .0        .0        .037
.0        .0        .0        .0        .0        .01
.0        .0        .0        .003      .0        .044
.0        .0        .0        .0        .01       .061
.003      .0        .0        .0        .013      .367
.003      .0        .0        .0        .003      .013
.0        .0        .0        .0        .0        .0
.003      .0        .0        .0        .0        .0
.0        .0        .0        .0        .0        .0
.007      .0        .0        .0        .003      .0
.0        .0        .0        .0        .0        .003
.007      .0        .0        .003      .0        .01
.0        .0        .0        .0        .0        .003
.007      .0        .0        .0        .0        .01
.0        .0        .0        .0        .007      .034
.0        .003      .010      .013      .034      .219
1.        2.        3.        4.        5.        6.        7.        8.
9.        10.       11.
304.      457.      610.      762.      914.      1372.
.0        .0        .0        .48       .13       .39
1.        .7        1.
.0        .3        .0
.0        .0        .0
.043      .957
OPTWDS,W,14,21
1,2,JUNE
1,2,J

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D

(PAGE 14 OF 15)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

2,2,DECEMBER
2,2,D
3,2,MARCH
3,2,M
4,2,SEPTEMBER
4,2,S
1,3,1400
1,3,14
2,3,0600
2,3,06
3,3,2200
3,3,22
REPORT RECORD LIST
INTERPRET PEDATA

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```



LISTING C-2 SAMPLE INPUT FOR PREPMS - CONT'D

(PAGE 15 OF 15)

```
JOB CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
ACCOUNT CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
COMMENT.  THIS RUNSTREAM IS USED WITH RUNS OF THE PREPMS PROGRAM MADE
COMMENT.  AFTER TAPE11 HAS BEEN CREATED.
COMMENT.  THE NEXT STATEMENT DECLARES TAPE5 TO BE A FILE OF RECORD
COMMENT.  TYPE Z WITH 80 CHARACTER LINES.  THIS IS REQUIRED SINCE TAPE5
COMMENT.  IS A FILE FROM THE "FRONT END" COMPUTER WHICH USES A DIFFERENT
COMMENT.  DEFAULT FILE TYPE FROM THE "BACK END" COMPUTER ON WHICH THIS
COMMENT.  JOB IS TO BE RUN.
FILE,TAPE5,RT=Z,FL=80,BT=C.
COMMENT.  THE NEXT STATEMENT OBTAINS A COPY OF THE PREPMS INPUT FROM THE
COMMENT.  "FRONT END" COMPUTER AND PUTS IT IN THE LOCAL FILE TAPE5.
GETPF,TAPE5,PREPMSSAMPLEINPUT,ID=SANDMEYER,ST=MFA.
COMMENT.  THE NEXT STATEMENT ATTACHES THE MACHINE LANGUAGE ("OBJECT
COMMENT.  CODE") CREATED BY RUNNING THE PREPMS SOURCE CODE THROUGH THE
COMMENT.  FORTRAN COMPILER IN A PREVIOUS JOB.
ATTACH,LGO,PREPMSSAMPLEBINARY,ID=SANDMEYER.
COMMENT.  THE NEXT STATEMENT ATTACHES THE ALREADY EXISTING TAPE11 ("CORE
COMMENT.  DATA BASE" FILE) WHICH WILL BE MODIFIED ("EXTENDED") BY THIS
COMMENT.  RUN.  THE MULTI-READ OPTION IS TURNED OFF SO THAT THE FILE CAN
COMMENT.  BE EXTENDED.
ATTACH,TAPE11,SAMPLETAPE11,ID=SANDMEYER,MR=0.
COMMENT.  THE NEXT STATEMENT LOADS AND EXECUTES THE PREPMS PROGRAM USING
COMMENT.  TAPE5 AS THE INPUT FILE.
LGO(TAPE5)
COMMENT.  THE NEXT STATEMENT EXTENDS TAPE11.  THIS MEANS THAT THE
COMMENT.  CHANGES MADE BY THIS RUN ARE NOW INCORPORATED IN THE PERMANENT
COMMENT.  FILE THAT WAS ATTACHED UNDER LOCAL FILE NAME TAPE11.
EXTEND,TAPE11.
COMMENT.  END OF JOB.
```

LISTING C-3  SAMPLE RUNSTREAM FOR PREPMS

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

**PREPMS**  THIS IS A CREATION RUN FOR RECORD OPTIMUM
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW RECORD,AINVDA0128,8
 RECORD NAMED : AINVDA0128 OF TYPE  8 NOW BEING READ
   16
    0.0000      .9900      .9500      .9000      .8000      .9000      .6000
  500.0000      .9500      .9000      .9500      .7500      .8500      .5500
 1000.0000      .9500      .8500      .9000      .7000      .8000      .4000
 1500.0000      .9000      .8000      .8500      .6000      .6000      .3000
 2000.0000      .8500      .8000      .8500      .6500      .7500      .3000
 2500.0000      .8000      .7500      .7000      .6000      .7000      .1500
 3000.0000      .7500      .7000      .6500      .5000      .6000      .0500
 3500.0000      .7000      .6500      .6000      .4500     0.0000     0.0000
 4000.0000      .6500      .6000      .5500      .4000     0.0000     0.0000
 4500.0000      .6000      .5500      .5000      .3500     0.0000     0.0000
 5000.0000      .5500      .5000      .4500      .3000     0.0000     0.0000
 5500.0000      .5000      .4500      .4000      .2500     0.0000     0.0000
 6000.0000      .4500      .4000      .3500      .2000     0.0000     0.0000
 6500.0000      .4000      .3500      .3000      .1500     0.0000     0.0000
 7000.0000      .3500      .3000      .2500      .1000     0.0000     0.0000
 9000.0000      .3000      .2000      .1000     0.0000     0.0000     0.0000
    0.0000    30.0000    90.0000   150.0000   300.0000  9000.0000
     .5000      .7000      .5000      .6000      .7000      .9500      .6000
     .5000      .6000      .4500      .7500      .7000     1.0000      .3000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW RECORD,JUN0600021,1
 RECORD NAMED : JUN0600021 OF TYPE  1 NOW BEING READ
    6   11
     .5590      .2200      .6900
     .0070      .0040     0.0000      .0040      .0220     0.0000      .0180      .0040
     .0510      .0320      .3430
     .0110     0.0000     0.0000      .0040      .0040      .0040      .0110      .0140
    0.0000      .0220      .4480
     .0180     0.0000     0.0000     0.0000     0.0000      .0150
     .0070     0.0000     0.0000     0.0000     0.0000      .0040
     .0040     0.0000     0.0000     0.0000     0.0000      .0110
    0.0000      .0070     0.0000     0.0000     0.0000      .0110
     .0040      .0040     0.0000     0.0000     0.0000      .0360
    0.0000     0.0000     0.0000     0.0000     0.0000     0.0000
     .0040     0.0000     0.0000     0.0000     0.0000      .0330
    0.0000     0.0000     0.0000     0.0000      .0040      .0040
    0.0000     0.0000      .0040     0.0000     0.0000      .0510
    0.0000     0.0000      .0040     0.0000     0.0000      .0360
     .0040      .0070      .0110      .0150      .0110      .2520
     .0040      .0070      .0040      .0040      .0040      .0150
    0.0000     0.0000     0.0000     0.0000     0.0000      .0070
     .0070     0.0000      .0070     0.0000     0.0000      .0070
     .0070     0.0000     0.0000     0.0000     0.0000     0.0000
     .0040     0.0000     0.0000     0.0000     0.0000      .0070
    0.0000     0.0000     0.0000     0.0000     0.0000     0.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4  SAMPLE OUTPUT FROM PREPMS

(PAGE 1 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    0.0000     0.0000      .0040    0.0000    0.0000      .0220
     .0070     0.0000      .0040    0.0000    0.0000      .0110
    0.0000     0.0000      .0040     .0040    0.0000      .0220
    0.0000      .0040     0.0000    0.0000    0.0000      .0220
     .0110      .0070      .0040     .0040     .0180      .2120
    1.0000     2.0000     3.0000    4.0000    5.0000    6.0000     7.0000     8.0000
    9.0000    10.0000    11.0000
  304.0000   457.0000   610.0000  762.0000  914.0000 1372.0000
    0.0000     0.0000     0.0000     .2700     .1200     .6100
    1.0000      .8800     1.0000
    0.0000      .1200     0.0000
    0.0000     0.0000     0.0000
    0.0000     1.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,JUN1400021,1
 RECORD NAMED : JUN1400021 OF TYPE  1 NOW BEING READ
    6   11
     .4690      .2200      .6900
    0.0000     0.0000     0.0000     .0040     .0040    0.0000      .0180     0.0000
     .0110      .0180      .4150
     .0110     0.0000     0.0000    0.0000     .0040    0.0000      .0040      .0140
     .0070     0.0000      .4910
    0.0000     0.0000     0.0000    0.0000    0.0000    0.0000
    0.0000     0.0000     0.0000    0.0000    0.0000    0.0000
    0.0000     0.0000     0.0060    0.0000    0.0000    0.0000
     .0040     0.0000     0.0000    0.0000    0.0000    0.0000
    0.0000     0.0000     0.0000    0.0000     .0040    0.0000
    0.0000     0.0000     0.0000    0.0000    0.0000    0.0000
     .0040      .0040      .0040    0.0000    0.0000     .0070
    0.0000     0.0000     0.0000    0.0000    0.0000    0.0000
    0.0000     0.0000      .0040     .0070    0.0000    0.0000
    0.0000     0.0000     0.0000    0.0000     .0040     .0140
     .0070      .0040      .0220     .0140     .1120     .2560
     .0070     0.0000     0.0000    0.0000    0.0000     .0040
    0.0000     0.0000     0.0000    0.0000    0.0000    0.0000
    0.0000     0.0000     0.0000    0.0000    0.0000    0.0000
    0.0000     0.0000     0.0000    0.0000    0.0000    0.0000
    0.0000      .0040     0.0000    0.0000    0.0000    0.0000
    0.0000     0.0000     0.0000    0.0000    0.0000    0.0000
    0.0000     0.0000     0.0000    0.0000     .0040    0.0000
     .0070      .0040     0.0000    0.0000     .0040    0.0000
    0.0000     0.0000     0.0000     .0040     .0040    0.0000
    0.0000     0.0000     0.0000    0.0000    0.0000    0.0000
     .0040      .0040      .0400     .0360     .2130     .1950
    1.0000     2.0000     3.0000    4.0000    5.0000    6.0000     7.0000     8.0000
    9.0000    10.0000    11.0000
  304.0000   457.0000   610.0000  762.0000  914.0000 1372.0000
     .1500      .2200      .2600     .3700    0.0000    0.0000
    1.0000      .3900     1.0000
    0.0000      .3600     0.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 2 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    0.0000     .2500    0.0000
     .6300     .3700
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * 
NEW REC,JUN2200021,1
 RECORD NAMED : JUN2200021 OF TYPE  1 NOW BEING READ
    6   11
     .5200     .2200     .6900
     .0100    0.0000    0.0000    0.0000     .0170    0.0000     .0280     .0170
     .0350     .0350     .3780
     .0030    0.0000    0.0000     .0100     .0030    0.0000     .0140     .0030
     .0100     .0280     .4060
     .0030    0.0000    0.0000    0.0000    0.0000     .0070
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000     .0070     .0070     .0030
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000     .0030    0.0000    0.0000     .0030     .0210
    0.0000     .0030    0.0000     .0030     .0030     .0070
    0.0000    0.0000     .0030     .0030     .0070     .0210
    0.0000    0.0000    0.0000    0.0000    0.0000     .0350
     .0030     .0030     .0100     .0030     .0170     .3390
    0.0000     .0030    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0030    0.0000    0.0000    0.0000     .0030     .0030
     .0030    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070    0.0000    0.0000    0.0000     .0030     .0030
    0.0000    0.0000    0.0000    0.0000    0.0000     .0030
    0.0000    0.0000     .0030    0.0000    0.0000     .0070
    0.0000    0.0000    0.0000     .0030    0.0000     .0240
    0.0000    0.0000     .0070     .0140     .0280     .3570
    1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
    9.0000   10.0000   11.0000
  304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
    0.0000    0.0000    0.0000     .1000     .0500     .8500
    1.0000    1.0000    1.0000
    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000
     .0500     .9500
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * 
NEW REC,DEC0600021,1
 RECORD NAMED : DEC0600021 OF TYPE  1 NOW BEING READ
    6   11
     .7350     .2200     .6900
     .0730     .0260     .0400     .0330     .1360    0.0000     .1030     .0170
     .0660     .0430     .1790
     .0260     .0100     .0030    0.0000     .0130    0.0000     .0170     .0230
     .0070     .0260     .1390

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4  SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 3 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

     .0400     .0070     .0070    0.0000     .0030     .0170
     .0200    0.0000     .0030    0.0000    0.0000     .0030
     .0230     .0030     .0030    0.0000    0.0000     .0100
     .0200     .0030    0.0000    0.0000    0.0000     .0100
     .0460     .0100     .0200     .0170     .0130     .0300
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0430     .0030     .0100     .0070     .0170     .0230
     .0030     .0030    0.0000    0.0000    0.0000     .0100
     .0170     .0030     .0100    0.0000     .0030     .0330
     .0100     .0030    0.0000    0.0000    0.0000     .0300
     .0070    0.0000     .0070     .0170     .0200     .1490
     .0100     .0030     .0030     .0030     .0030     .0030
     .0100    0.0000    0.0000    0.0000    0.0000    0.0000
     .0030    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070    0.0000    0.0000    0.0000    0.0000     .0070
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070    0.0000    0.0000     .0070     .0030    0.0000
     .0200    0.0000    0.0000    0.0000     .0030    0.0000
    0.0000     .0030     .0030    0.0000    0.0000    0.0000
     .0100     .0070    0.0000    0.0000     .0030     .0070
     .0260     .0230     .0130     .0100     .0260     .0400
    1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
    9.0000   10.0000   11.0000
  304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
    0.0000    0.0000    0.0000     .6600     .0900     .2600
    1.0000     .6000    1.0000
    0.0000     .2000    0.0000
    0.0000     .2000    0.0000
     .0100     .9900
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,DEC1400021,1
 RECORD NAMED : DEC1400021 OF TYPE  1 NOW BEING READ
    6   11
     .6370     .2200     .6900
     .0500     .0200     .0230     .0270     .1040     .0030     .0670     .0400
     .0470     .0340     .2210
     .0200     .0030     .0070     .0130     .0070     .0100     .0230     .0130
     .0070     .0230     .2350
     .0340     .0030     .0030     .0030     .0070    0.0000
     .0100     .0100    0.0000    0.0000    0.0000    0.0000
     .0100    0.0000     .0030     .0030     .0030     .0030
     .0100     .0030    0.0000    0.0000    0.0000     .0130
     .0470     .0170    0.0000    0.0000     .0070     .0340
    0.0000    0.0000    0.0000    0.0000    0.0000     .0030
     .0340     .0070    0.0000    0.0000     .0200     .0070
     .0200     .0030     .0030    0.0000    0.0000     .0130
     .0100     .0030     .0070    0.0000     .0030     .0230
    0.0000     .0030     .0030     .0030    0.0000     .0230
     .0070     .0130     .0200     .0170     .0200     .1440

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 4 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

     .0070    0.0000     .0070     .0030    0.0000     .0030
     .0030    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070     .0030    0.0000    0.0000    0.0000     .0030
     .0030    0.0000    0.0000    0.0000    0.0000     .0030
     .0100    0.0000    0.0000    0.0000    0.0000    0.0000
     .0130     .0030     .0030    0.0000    0.0000     .0030
     .0030    0.0000    0.0000    0.0000     .0070     .0030
    0.0000    0.0000    0.0000    0.0000     .0030     .0030
     .0030     .0030    0.0000     .0030    0.0000     .0130
     .0340     .0370     .0340     .0340     .0340     .0640
    1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
    9.0000   10.0000   11.0000
  304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
    0.0000     .0500     .1600     .7900    0.0000    0.0000
    1.0000     .5300    1.0000
    0.0000     .2100    0.0000
    0.0000     .2600    0.0000
     .0200     .0800
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW PEC,DEC2200021,1
 RECORD NAMED : DEC2200021 OF TYPE  1 NOW BEING READ
    6   11
     .7570     .2200     .6900
     .0830     .0270     .0370     .0330     .1230    0.0000     .1070     .0130
     .0630     .0400     .2300
     .0230    0.0000     .0070     .0070     .0170    0.0000     .0270     .0130
     .0170     .0130     .1200
     .0330    0.0000     .0030     .0070     .0030     .0370
     .0230    0.0000     .0030    0.0000    0.0000    0.0000
     .0200    0.0000     .0100    0.0000    0.0000     .0070
     .0130    0.0000     .0030    0.0000     .0030     .0130
     .0500     .0200     .0070     .0100     .0200     .0170
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0300     .0170     .0070    0.0000     .0170     .0370
     .0030    0.0000    0.0000    0.0000     .0070     .0030
     .0100     .0100     .0070     .0030     .0070     .0270
     .0030    0.0000    0.0000     .0030     .0070     .0270
     .0070     .0100     .0270     .0230     .0300     .1330
     .0100     .0030    0.0000     .0100    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070    0.0000     .0030     .0030    0.0000     .0030
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070     .0030    0.0000    0.0000     .0030     .0130
     .0100    0.0000     .0030    0.0000    0.0000    0.0000
     .0070    0.0000    0.0000    0.0000    0.0000     .0100
    0.0000     .0030     .0070    0.0000    0.0000     .0030
     .0170     .0130     .0100     .0100     .0100     .0600

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4  SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 5 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
    9.0000   10.0000   11.0000
  304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
    0.0000    0.0000    0.0000     .8200     .0700     .1100
    1.0000     .4700    1.0000
    0.0000     .2500    0.0000
    0.0000     .2800    0.0000
    0.0000    1.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,ACQDAT0122,2
 RECORD NAMED : ACQDAT0122 OF TYPE  2 NOW BEING READ
   11   11    5
    0.0000     .1000     .2000     .3000     .4000     .5000     .6000     .7000
     .8000     .9000    1.0000
    0.0000  900.0000 1500.0000 2300.0000 2900.0000 3400.0000 3900.0000 4300.0000
 4600.0000 4900.0000 7000.0000
    0.0000     .1000     .2000     .3000     .4000     .5000     .6000     .7000
     .8000     .9000    1.0000
   25.0000   40.0000   70.0000   90.0000  150.0000  230.0000  300.0000  420.0000
  580.0000 1140.0000 1400.0000
   25.0000   40.0000   70.0000  130.0000  230.0000  300.0000  360.0000  400.0000
  500.0000  670.0000  980.0000
   25.0000   40.0000   70.0000  130.0000  230.0000  400.0000  480.0000  620.0000
 1000.0000 1330.0000 1600.0000
   25.0000   40.0000   70.0000  130.0000  210.0000  320.0000  380.0000  420.0000
  460.0000  570.0000  980.0000
   25.0000   50.0000  100.0000  170.0000  250.0000  330.0000  390.0000  490.0000
  720.0000  950.0000 1425.0000
 1000.0000 1500.0000 2000.0000 2500.0000 9999.0000
    5.0000    3.0000
    3.0000    2.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,ACQDAT0222,2
 RECORD NAMED : ACQDAT0222 OF TYPE  2 NOW BEING READ
   11   11    3
    0.0000     .1000     .2000     .3000     .4000     .5000     .6000     .7000
     .8000     .9000    1.0000
    0.0000  400.0000  650.0000  900.0000 1300.0000 1600.0000 2000.0000 2500.0000
 3000.0000 3700.0000 7000.0000
    0.0000     .1000     .2000     .3000     .4000     .5000     .6000     .7000
     .8000     .9000    1.0000
   25.0000   30.0000   40.0000   50.0000   70.0000  120.0000  140.0000  180.0000
  280.0000  500.0000 1100.0000
   25.0000   30.0000   40.0000   50.0000   70.0000  120.0000  150.0000  200.0000
  420.0000  900.0000 1520.0000
   25.0000   40.0000   60.0000  100.0000  140.0000  190.0000  210.0000  240.0000
  310.0000  590.0000 1600.0000
 2000.0000 2500.0000 9999.0000
    3.0000    2.0000
    2.0000    2.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 6 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,ACQDAT0322,2
 RECORD NAMED : ACQDAT0322 OF TYPE  2 NOW BEING READ
    8   11    1
    0.0000     .0200     .0400     .1000     .2000     .3000     .3200    1.0000
    0.0000 2000.0000 3000.0000 3500.0000 5000.0000 5900.0000 7000.0000 9999.0000
    0.0000     .1000     .2000     .3000     .4000     .5000     .6000     .7000
     .8000     .9000    1.0000
   25.0000   40.0000   70.0000  130.0000  230.0000  400.0000  480.0000  620.0000
 1000.0000 1330.0000 1600.0000
 9999.0000
    8.0000    3.0000
    5.0000    2.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,PSPDAT0223,3
 RECORD NAMED : PSPDAT0223 OF TYPE  3 NOW BEING READ
    9
    0.0000     .1250     .2500     .3750     .5000     .6250     .7500     .8750
    1.0000
    0.0000    5.0000   10.0000   18.0000   24.0000   36.0000   52.0000   87.0000
  174.0000
   45.0000   45.0000   20.0000   54.0000   97.0000
   17.3700   37.3700   27.6700   47.6700    6.0000   25.0000    6.0000   73.0000
   13.5700   15.7200   32.9200   37.1200    6.9000    6.9000    5.9500    5.9500
   89.0000  112.0000   57.0000   89.0000  176.0000   89.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,PSTDAT0127,7
 RECORD NAMED : PSTDAT0127 OF TYPE  7 NOW BEING READ
    8
    0.0000  500.0000 1000.0000 1500.0000 2000.0000 2500.0000 3000.0000 9000.0000
     .0700     .0700     .1300     .1500     .1600     .1700     .1800     .1800
     .9000     .9000     .8200     .7800     .7600     .7400     .7300     .7300
     .0300     .0300     .0500     .0700     .0800     .0900     .0900     .0900
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,DESDAT0124,4
 RECORD NAMED : DESDAT0124 OF TYPE  4 NOW BEING READ
    8
    0.0000  500.0000 1000.0000 1500.0000 2000.0000 3000.0000 3001.0000 9999.0000
    0.0000     .5000     .4000     .3000     .2000     .1000    0.0000    0.0000
    1.0000     .3000     .2000     .0500     .0100    0.0000    0.0000    0.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,DESDAT0224,4
 RECORD NAMED : DESDAT0224 OF TYPE  4 NOW BEING READ
    2
    0.0000 9999.0000
    0.0000    0.0000
    0.0000    0.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTION WORDS,NEXTCASE,4,2
 OPTION WORD TYPE RECORD NAMED : NEXTCASE    BEING INPUT

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 7 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

 4 OPTION CARDS READ FOR KEYWORD NUMBER  2
1,2,RESETDEFAULTS
1,2,RD
2,2,DON'T RESET
2,2,DR
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPT WDS,PLOT,4,5
 OPTION WORD TYPE RECORD NAMED : PLOT       BEING INPUT
 4 OPTION CARDS READ FOR KEYWORD NUMBER  5
1,2,PRINTER
2,2,CALCOMP
1,3,OFF
2,3,ON
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPT WDS,LOSMODE,4,11
 OPTION WORD TYPE RECORD NAMED : LOSMODE    BEING INPUT
 4 OPTION CARDS READ FOR KEYWORD NUMBER 11
1,2,SHOOTINGGALLERY
1,2,SG
2,2,RANDOMOCCURRENCE
2,2,RO
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPT WDS,MSNCODE,6,12
 OPTION WORD TYPE RECORD NAMED : MSNCODE    BEING INPUT
 6 OPTION CARDS READ FOR KEYWORD NUMBER 12
1,2,PRIORITYPREPLANNEDTARGET
1,2,PRPT
2,2,PREPLANNEDTARGET
2,2,PPT
3,2,TARGETOFOPPORTUNITY
3,2,TOO
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,NUMDIFTGTP,4,13
 OPTION WORD TYPE RECORD NAMED : NUMDIFTGTP BEING INPUT
 4 OPTION CARDS READ FOR KEYWORD NUMBER 13
1,2,THREE
1,2,3
2,2,TWO
2,2,2
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,TARGETTYPE,7,14
 OPTION WORD TYPE RECORD NAMED : TARGETTYPE BEING INPUT
 7 OPTION CARDS READ FOR KEYWORD NUMBER 14
1,2,1
2,2,2
3,2,3
4,2,4
5,2,5
6,2,6
7,2,7

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 8 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,DESIGTYPE,6,16
 OPTION WORD TYPE RECORD NAMED : DESIGTYPE  BEING INPUT
 6 OPTION CARDS READ FOR KEYWORD NUMBER 16
1,2,GLLD
1,2,G
2,2,MULF
2,2,M
3,2,LTD
3,2,L
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,FOPOSITION,4,17
 OPTION WORD TYPE RECORD NAMED : FOPOSITION BEING INPUT
 4 OPTION CARDS READ FOR KEYWORD NUMBER 17
1,2,VANTAGEPOINT
1,2,VP
2,2,MANEUVERUNIT
2,2,MU
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,TLEFACTORS,1,33
 OPTION WORD TYPE RECORD NAMED : TLEFACTORS BEING INPUT
 1 OPTION CARDS READ FOR KEYWORD NUMBER 33
1,1,NOTHING
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,ACQRNGDIST,8,22
 OPTION WORD TYPE RECORD NAMED : ACQRNGDIST BEING INPUT
 8 OPTION CARDS READ FOR KEYWORD NUMBER 22
1,2,AVERAGE
1,2,A
2,2,CLOSE
2,2,C
3,2,OPEN
3,2,O
4,2,STACS
4,2,S
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,RESPTIME,15,23
 OPTION WORD TYPE RECORD NAMED : RESPTIME   BEING INPUT
15 OPTION CARDS READ FOR KEYWORD NUMBER 23
2,2,STAUCH
2,2,S
1,2,FTSILL
1,2,FS
3,2,SCUNGIU
11,2,PAPAM
11,2,P
1,3,DIGITAL
1,3,D
2,3,VOICE
2,3,V

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4  SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 9 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

1,4,DAY
1,4,D
2,4,NIGHT
2,4,N
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPT WDS,DFIRESUPP,4,24
 OPTION WORD TYPE RECORD NAMED : DFIRESUPP BEING INPUT
 4 OPTION CARDS READ FOR KEYWORD NUMBER 24
1,2,HIGH
1,2,H
2,2,NONE
2,2,N
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,PUNC,2,31
 OPTION WORD TYPE RECORD NAMED : PUNC       BEING INPUT
 2 OPTION CARDS READ FOR KEYWORD NUMBER 31
1,2,NO
1,2,N
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,CONTROL,3,9
 OPTION WORD TYPE RECORD NAMED : CONTROL    BEING INPUT
 3 OPTION CARDS READ FOR KEYWORD NUMBER  9
1,2,ECHO
1,3,SHORT
2,3,LONG
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,SMOKE,4,32
 OPTION WORD TYPE RECORD NAMED : SMOKE      BEING INPUT
 4 OPTION CARDS READ FOR KEYWORD NUMBER 32
1,2,SMOKEROUNDSFIRED
1,2,SMKRNDS
2,2,PROBABILITYSMOKEKILLSMISSION
2,2,PRBSMOKE
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RSPDAT0123,3
 RECORD NAMED : RSPDAT0123 OF TYPE  3 NOW BEING READ
    9
    0.0000     .1250     .2500     .3750     .5000     .6250     .7500     .8750
    1.0000
    0.0000    5.0000   10.0000   18.0000   24.0000   36.0000   52.0000   87.0000
  174.0000
   20.0000   20.0000   20.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
   69.1900   69.1900   69.1900   69.1900   46.3700   46.3200   46.3200   46.3200
   75.0000   75.0000   75.0000   75.0000   75.0000   75.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RODAV3TPH1,5
 RECORD NAMED : RODAV3TPH1 OF TYPE  5 NOW BEING READ
    5
   15.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 10 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

 1000.0000 1500.0000 2000.0000 2500.0000 9999.0000
     .7392     .7040     .8008     .6962     .7617
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RODAV5TPN1,5
 RECORD NAMED : RODAV5TPN1 OF TYPE  5 NOW BEING READ
    5
   15.0000
 1000.0000 1500.0000 2000.0000 2500.0000 9999.0000
     .6831     .6403     .7580     .6309     .7078
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RODAV8TPN1,5
 RECORD NAMED : RODAV8TPN1 OF TYPE  5 NOW BEING READ
    5
   15.0000
 1000.0000 1500.0000 2000.0000 2500.0000 9999.0000
     .6129     .5539     .6998     .5422     .6336
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RODAV3TPN2,5
 RECORD NAMED : RODAV3TPN2 OF TYPE  5 NOW BEING READ
    3
   15.0000
 2000.0000 2500.0000 9999.0000
     .5749     .6702     .6599
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RODAV5TPN2,5
 RECORD NAMED : RODAV5TPN2 OF TYPE  5 NOW BEING READ
    5
   15.0000
 2000.0000 2500.0000 9999.0000    0.0000    0.0000
     .5023     .6139     .5888    0.0000    0.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RODAV8TPN2,5
 RECORD NAMED : RODAV8TPN2 OF TYPE  5 NOW BEING READ
    3
   15.0000
 2000.0000 2500.0000 9999.0000
     .4114     .5434     .4959
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RODAV3TPN3,5
 RECORD NAMED : RODAV3TPN3 OF TYPE  5 NOW BEING READ
    1
   15.0000
 9999.0000
     .8008
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RODAV5TPN3,5
 RECORD NAMED : RODAV5TPN3 OF TYPE  5 NOW BEING READ
    1
   15.0000
 9999.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 11 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

     .7580
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RODAVBTPN3,5
 RECORD NAMED : RODAVBTPN3 OF TYPE  5 NOW BEING READ
    1
   15.0000
 9999.0000
     .6998
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEW REC,RSPDAT0323,3
 RECORD NAMED : RSPDAT0323 OF TYPE  3 NOW BEING READ
    9
    0.0000     .1250     .2500     .3750     .5000     .6250     .7500     .8750
    1.0000
    0.0000    5.0000   10.0000   18.0000   24.0000   36.0000   52.0000   87.0000
  174.0000
    0.0000    0.0000    0.0000    0.0000    0.0000
   50.0000   50.0000   50.0000   50.0000    0.0000    0.0000    0.0000    0.0000
  159.0000  159.0000  159.0000  159.0000   80.0000   80.0000   80.0000   80.0000
  209.0000  209.0000  209.0000  209.0000  209.0000  209.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEWREC,MAP0600021,1
 RECORD NAMED : MAP0600021 OF TYPE  1 NOW BEING READ
    6   11
     .6590     .2200     .6900
     .0320     .0100     .0230     .0390     .0420    0.0000     .0710     .0100
     .0870     .0680     .2800
     .0260     .0100    0.0000     .0190    0.0000    0.0000     .0290     .0190
     .0350     .0290     .1740
     .0130    0.0000    0.0000    0.0000     .0030     .0160
     .0100    0.0000    0.0000    0.0000    0.0000    0.0000
     .0060     .0060     .0060    0.0000    0.0000     .0030
     .0060    0.0000     .0100     .0030     .0030     .0160
     .0060     .0100    0.0000    0.0000    0.0000     .0260
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0130     .0130    0.0000     .0060     .0060     .0320
    0.0000     .0030    0.0000    0.0000     .0030     .0030
     .0130     .0100     .0030     .0060     .0130     .0420
     .0060     .0060     .0100    0.0000    0.0000     .0450
    0.0000    0.0000     .0160     .0130     .0260     .2250
     .0030    0.0000     .0030    0.0000    0.0000     .0190
     .0060    0.0000    0.0000    0.0000    0.0000     .0030
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0060    0.0000     .0030    0.0000    0.0000     .0100
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0060    0.0000    0.0000    0.0000    0.0000     .0230
     .0030    0.0000    0.0000     .0030    0.0000     .0130
     .0060     .0030    0.0000    0.0000     .0030     .0230
    0.0000     .0030    0.0000    0.0000     .0030     .0230

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 12 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    .0030     .0030     .0160    0.0000     .0260     .1250
   1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
   9.0000   10.0000   11.0000
 304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
   0.0000    0.0000    0.0000     .6500     .1100     .2400
   1.0000     .5100    1.0000
   0.0000     .3000    0.0000
   0.0000     .1900    0.0000
    .0090     .9910
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * 
NEWREC,MAP1400021,1
 RECORD NAMED : MAP1400021 OF TYPE  1 NOW BEING READ
   6   11
    .5980     .2200     .6900
    .0090     .0030     .0060     .0030     .0320     .0030     .0190     .0090
    .0250     .0600     .4180
   0.0000    0.0000     .0030     .0060     .0030    0.0000     .0090     .0160
    .0220     .0000     .3320
    .0090    0.0000    0.0000    0.0000    0.0000    0.0000
   0.0000     .0030    0.0000    0.0000    0.0000    0.0000
    .0030    0.0000    0.0000    0.0000    0.0000     .0030
   0.0000    0.0000    0.0000    0.0000     .0030    0.0000
    .0090     .0090     .0030     .0030    0.0000     .0060
   0.0000    0.0000    0.0000    0.0000    0.0000     .0030
    .0030     .0030    0.0000    0.0000     .0030     .0090
   0.0000    0.0000     .0030    0.0000    0.0000     .0060
    .0030    0.0000     .0130    0.0000     .0030     .0160
   0.0000     .0030     .0030     .0090     .0130     .0320
    .0190     .0130     .0280     .0510     .0730     .2340
   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    .0030    0.0000    0.0000    0.0000    0.0000    0.0000
    .0030    0.0000     .0030    0.0000    0.0000    0.0000
   0.0000    0.0000    0.0000    0.0000     .0030    0.0000
   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    .0060    0.0000    0.0000    0.0000     .0030    0.0000
   0.0000    0.0000    0.0000     .0030    0.0000     .0130
   0.0000     .0090     .0060     .0030    0.0000     .0030
   0.0000     .0030    0.0000    0.0000    0.0000     .0060
    .0060     .0130     .0250     .0600     .1080     .1200
   1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
   9.0000   10.0000   11.0000
 304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
    .0400     .1500     .2100     .6000    0.0000    0.0000
   1.0000     .3000    1.0000
   0.0000     .2800    0.0000
   0.0000     .4200    0.0000
    .3770     .6230
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * 
NEWREC,MAP2200021,1

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 13 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

 RECORD NAMED : MAR2200021 OF TYPE  1 NOW BEING READ
    6   11
     .6790     .2200     .6900
     .0160     .0030     .0060     .0030     .0420     .0030     .0320     .0190
     .0770     .0870     .3910
     .0100    0.0000    0.0000     .0030     .0030    0.0000     .0320     .0100
     .0160     .0220     .2240
    0.0000    0.0000    0.0000    0.0000    0.0000     .0160
     .0030    0.0000    0.0000    0.0000    0.0000    0.0000
     .0030     .0030    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000     .0030
     .0100     .0030     .0030     .0060     .0030     .0160
    0.0000    0.0000    0.0000    0.0000    0.0000     .0030
    0.0000    0.0000    0.0000     .0030    0.0000     .0290
    0.0000    0.0000    0.0000     .0030    0.0000     .0160
     .0060     .0100     .0100     .0030     .0030     .0450
    0.0000    0.0000     .0060    0.0000     .0060     .0740
    0.0000     .0030     .0130     .0130     .0380     .3240
    0.0000    0.0000    0.0000     .0030    0.0000     .0060
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0030    0.0000    0.0000    0.0000    0.0000    0.0000
     .0030    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0220    0.0000    0.0000    0.0000     .0030     .0060
    0.0000     .0030    0.0000    0.0000    0.0000     .0060
    0.0000     .0100    0.0000     .0030    0.0000     .0030
    0.0000     .0030    0.0000     .0030     .0060     .0100
     .0060     .0030     .0130     .0060     .0160     .1790
    1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
    9.0000   10.0000   11.0000
  304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
    0.0000    0.0000    0.0000     .5800     .0800     .3400
    1.0000     .4200    1.0000
    0.0000     .2700    0.0000
    0.0000     .3100    0.0000
    0.0000    1.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * 
NEWREC,SEP0500021,1
 RECORD NAMED : SEP0600021 OF TYPE  1 NOW BEING READ
    6   11
     .5680     .2200     .6900
     .1290     .0250     .0360     .0220     .0430    0.0000     .0400    0.0000
     .0500     .0400     .1830
     .0400     .0070     .0140     .0180     .0320    0.0000     .0320     .0110
     .0360     .0220     .2190
     .1010    0.0000    0.0000    0.0000    0.0000     .0290
     .0070    0.0000    0.0000     .0040    0.0000     .0140
     .0040    0.0000     .0070    0.0000    0.0000     .0250
    0.0000    0.0000    0.0000    0.0000    0.0000     .0220

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 14 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000011111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

   .0040    0.0000    0.0000    0.0000     .0040     .0360
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  0.0000    0.0000    0.0000    0.0000     .0040     .0360
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  0.0000     .0040    0.0000    0.0000     .0070     .0400
  0.0000    0.0000    0.0000    0.0000     .0040     .0360
  0.0000     .0040     .0040     .0110     .0180     .1470
   .0140    0.0000    0.0000    0.0000    0.0000     .0250
   .0070    0.0000    0.0000    0.0000    0.0000    0.0000
  0.0000     .0040    0.0000    0.0000    0.0000     .0110
   .0040    0.0000    0.0000    0.0000    0.0000     .0140
   .0110    0.0000    0.0000     .0040    0.0000     .0180
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  0.0000    0.0000    0.0000    0.0000    0.0000     .0320
  0.0000    0.0000    0.0000    0.0000    0.0000     .0110
   .0040    0.0000    0.0000    0.0000     .0040     .0290
  0.0000     .0040    0.0000     .0040     .0070     .0070
   .0220     .0070     .0110     .0040     .0220     .1550
  1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
  9.0000   10.0000   11.0000
304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
  0.0000    0.0000    0.0000     .5700     .1100     .3200
  1.0000     .7900    1.0000
  0.0000     .1500    0.0000
  0.0000     .0600    0.0000
  0.0000    1.0000
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEWREC,SEP1400021,1
 RECORD NAMED : SEP1400021 OF TYPE  1 NOW BEING READ
   6   11
   .4370     .2200     .6900
  0.0000    0.0000    0.0000     .0070     .0100     .0030     .0140    0.0000
   .0140     .0310     .3570
   .0070    0.0000    0.0000    0.0000     .0100    0.0000     .0100     .0210
   .0030     .0240     .4880
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
   .0030     .0030    0.0000    0.0000    0.0000    0.0000
   .0030    0.0000    0.0000    0.0000    0.0000     .0070
  0.0000    0.0000    0.0000    0.0000    0.0000     .0030
  0.0000    0.0000    0.0000    0.0000     .0300     .0100
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  0.0000    0.0000    0.0000     .0030     .0030     .0070
  0.0000    0.0000     .0030     .0030    0.0000     .0240
   .0030     .0030     .0070     .0070     .0450     .2920
   .0030    0.0000    0.0000    0.0000    0.0000     .0030
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000011111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4  SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 15 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

     .0030     .0030    0.0000    0.0000    0.0000     .0030
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0030    0.0000    0.0000    0.0000     .0070    0.0000
     .0030    0.0000    0.0000     .0100     .0030     .0030
    0.0000    0.0000    0.0000    0.0000    0.0000     .0030
    0.0000    0.0000     .0070    0.0000     .0070     .0100
     .0070     .0170     .0310     .0340     .2200     .1790
    1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
    9.0000   10.0000   11.0000
  304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
     .0400     .2100     .2900     .4600    0.0000    0.0000
    1.0000     .4300    1.0000
    0.0000     .3200    0.0000
    0.0000     .2500    0.0000
     .5470     .4530
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
NEWREC,SEP2200021,1
 RECORD NAMED : SEP2200021 OF TYPE  1 NOW BEING READ
    6   11
     .6020     .2200     .6900
     .0130    0.0000     .0030     .0100     .0240    0.0000     .0400     .0100
     .0470     .0710     .3840
     .0200    0.0000     .0030    0.0000     .0100     .0030     .0200     .0030
     .0170     .0400     .2790
    0.0000    0.0000    0.0000    0.0000    0.0000     .0130
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000     .0030    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000     .0100
     .0030    0.0000    0.0000    0.0000    0.0000     .0200
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0030    0.0000    0.0000    0.0000    0.0000     .0370
    0.0000    0.0000    0.0000    0.0000    0.0000     .0100
    0.0000    0.0000    0.0000     .0030    0.0000     .0440
    0.0000    0.0000    0.0000    0.0000     .0100     .0610
     .0030    0.0000    0.0000    0.0000     .0130     .3670
     .0030    0.0000    0.0000    0.0000     .0030     .0130
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0030    0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
     .0070    0.0000    0.0000    0.0000     .0030    0.0000
    0.0000    0.0000    0.0000    0.0000    0.0000     .0030
     .0070    0.0000    0.0000     .0030    0.0000     .0100
    0.0000    0.0000    0.0000    0.0000    0.0000     .0030
     .0070    0.0000    0.0000    0.0000    0.0000     .0100
    0.0000    0.0000    0.0000    0.0000     .0070     .0340
    0.0000     .0030     .0100     .0130     .0340     .2190
    1.0000    2.0000    3.0000    4.0000    5.0000    6.0000    7.0000    8.0000
    9.0000   10.0000   11.0000
  304.0000  457.0000  610.0000  762.0000  914.0000 1372.0000
    0.0000    0.0000    0.0000     .4800     .1300     .3900

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4 SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 16 OF 17)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    1.0000     .7000    1.0000
    0.0000     .3000    0.0000
    0.0000    0.0000    0.0000
     .0430     .9570
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
OPTWDS,W,14,21
 OPTION WORD TYPE RECORD NAMED : W           BEING INPUT
14 OPTION CARDS READ FOR KEYWORD NUMBER 21
1,2,JUNE
1,2,J
2,2,DECEMBER
2,2,D
3,2,MARCH
3,2,M
4,2,SEPTEMBER
4,2,S
1,3,1400
1,3,14
2,3,0600
2,3,06
3,3,2200
3,3,22
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
REPORT RECORD LIST
 RECORD NAMES AND FILE INFORMATION FOR TAPE11
PEOOAIAAJS  20001024400000000101          PEOOAI:AJS  20001024400000010345          PEOOAIACZS  20001024400000020611
AINVDA0128  20000024700000031055          JUN0600021  20000030100000031324          JUN1400021  20000030100000031625
JUN2200021  20000030100000032126          DEC0600021  20000030100000032427          DEC1400021  20000030100000032730
DEC2200021  20000030100000033231          ACQDAT0122  20000024100000033532          ACQDAT0222  20000024100000033773
ACQDAT0322  20000024100000034234          PSPDAT0223  20000006100000034475          PSTDAT0127  20000005100000034556
DFSDAT0124  20000003700000034627          DFSDAT0224  20000003700000034666          NEXTCASE    20000045400000034725
OPTIONS     20000010600000035401          PLOT        20000045400000035507          LOSMODF     20000045400000036163
MSNCODE     20000045400000036637          NUMDIFTGTP  20000045400000037313          TARGETTYPE  20000045400000037767
DESIGTYPE   20000045400000040443          FOPOSITION  20000045400000041117          TLEFACTORS  20000045400000041573
ACQPNGDIST  20000045400000042247          RESPTIME    20000045400000042723          DFIRESUPPP  20000045400000043377
RUMC        20000045400000044053          CONTROL     20000045400000044527          SMOKE       20000045400000045203
PSPDAT0123  20000006100000045657          PODAV3TPN1  20000002600000045740          PODAV5TPN1  20000002600000045766
PODAVRTPN1  20000002600000046014          PODAV3TPN2  20000002600000046042          RODAV5TPN2  20000002600000046070
PODAVRTPN2  20000002600000046116          PODAV3TPN3  20000002600000046144          RODAV5TRN3  20000002600000046172
PODAVRTPN3  20000002600000046220          PSPDAT0323  20000006100000046246          MAP0600021  20000030100000046327
MAP1400021  20000030100000046630          MAP2200021  20000030100000047131          SEP0600021  20000030100000047432
SEP1400021  20000030100000047733          SEP2200021  20000030100000050234          W           20000045400000050535

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
INTERPRET PEDATA
    PE RECORD: PEOOAIAAJS TP=106. DT-1 VEL=5. GTP= 8. REFL= .10 ANGLET= 25. DEFB=  0. TGTHD= 0. SKSEN=60.
    PE RECORD: PEOOAI:AJS TP=106. DT-1 VEL-2. GTP= 8. REFL= .10 ANGLET= 25. DEFB=  0. TGTHD= 0. SKSEN=60.
    PE RECORD: PEOOAIACZS TP=106. DT-1 VEL=5. GTP= 8. REFL= .30 ANGLET= 60. DEFB=  0. TGTHD= 0. SKSEN=60.
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING C-4  SAMPLE OUTPUT FROM PREPMS - CONT'D

(PAGE 17 OF 17)



Next page is blank.

APPENDIX D

LISTING AND SAMPLE CASE OF COPE

# APPENDIX D

## LISTING AND SAMPLE CASE OF COPE

This appendix contains:

(1) A listing of the FORTRAN code for the COPE program.

(2) One set of sample case inputs for COPE.

(3) One sample runstream for COPE.

(4) Sample output created by running COPE with the sample input (and with the TAPE11 created by previous PAM and PREPMS sample case runs). The output produced on TAPE8 by this run (for cases 1 and 2) is included in Listings 7-1 and 7-2 of section 7.6.

Note: The column numbers included in some of the listings in this appendix are not a part of the program code, the data, or the output, but are included only for the reader's convenience.

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      PROGRAM COPE (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,DEBUG=OUTPUT, 001990  PP-COPE  2
     1      TAPE4,TAPE7,TAPE8,TAPE9,TAPE10,TAPE11)                       002000  PP-COPE  3
C                                                                        002010  PP-COPE  4
C     *     *     *     *     *     *     *     *     *     *            002020  PP-COPE  5
C                                                                        002030  PP-COPE  6
C     COPE TAPE (I/O UNIT) USES                                          002040  PP-COPE  7
C                                                                        002050  PP-COPE  8
C    -TAPE5 (INPUT) IS THE I/O UNIT FROM WHICH THE INPUT DIRECTIVES      002060  PP-COPE  9
C     ARE READ.  IT IS ALSO USED WHEN THE "TEMPORARY" OPTION IS USED TO  002070  PP-COPE 10
C     READ FORMATTED INPUT.                                              002080  PP-COPE 11
C    -TAPE4 IS AN ALTERNATE INPUT FILE THAT MAY BE USED WHEN READING PE  002090  PP-COPE 12
C     DATA UNDER THE "TEMPORARY" OPTION.                                 002100  PP-COPE 13
C     TAPE6 (OUTPUT) IS THE FILE TO WHICH THE NORMAL OUTPUT IS WRITTEN.  002110  PP-COPE 14
C     IT IS USUALLY THE FILE SENT TO THE PRINTER FOR HARD COPY OUTPUT.   002120  PP-COPE 15
C     IT IS ALSO EQUATED TO DEBUG SO THAT SUBROUTINE TRACEBACKS WILL BE  002130  PP-COPE 16
C     PRINTED ON IT.                                                     002140  PP-COPE 17
C    -TAPE7 IS NOT CURRENTLY USED.  IT IS INTENDED AS THE FILE TO WHICH  002150  PP-COPE 18
C     OUTPUT WILL BE WRITTEN FOR USE BY A POSSIBLE FUTURE COPE ANALYSIS  002160  PP-COPE 19
C     PROGRAM.                                                           002170  PP-COPE 20
C    -TAPE8 IS THE FILE TO WHICH ECHO PRINTS ("DATA CHECKS") ARE         002180  PP-COPE 21
C     WRITTEN.  UNLESS USER ACTION IS TAKEN, IT IS TREATED AS A SCRATCH  002190  PP-COPE 22
C     FILE.                                                              002200  PP-COPE 23
C    -TAPE9 IS THE FILE TO WHICH PRINTER PLOTS WILL BE WRITTEN IF        002210  PP-COPE 24
C     AND WHEN THAT PLOT OPTION IS IMPLEMENTED.                          002220  PP-COPE 25
C    -TAPE10 IS THE FILE TO WHICH THE CALCOMP PLOT OUTPUT WILL BE        002230  PP-COPE 26
C     WRITTEN IF AND WHEN THAT PLOT OPTION IS IMPLEMENTED.               002240  PP-COPE 27
C    -TAPE11 IS THE WORD ADDRESSABLE MASS STORAGE "DATA BASE" FILE FROM  002250  PP-COPE 28
C     WHICH COPE OBTAINS THE DATA RECORDS USED TO FILL THE EIGHT DATA    002260  PP-COPE 29
C     BLOCKS AS DESCRIBED IN THE DOCUMENTATION.   TAPE11 IS CREATED BY   002270  PP-COPE 30
C     THE PREPMS AND PAP PREPROCESSOR PROGRAMS.                          002280  PP-COPE 31
C                                                                        002290  PP-COPE 32
C     *     *     *     *     *     *     *     *     *     *            002300  PP-COPE 33
C                                                                        002310  PP-COPE 34
      COMMON /BAIL/ BLURBS, IBAIL                                        000170  CM*BAIL  2
      COMMON /DODF/ DFORKL(10,3), NDESP, NDOKIL, DOKILD                  000250  CM*DODF  2
      LOGICAL DOKILD                                                     000260  CM*DODF  3
      COMMON /FLAG/ ENDREP, IOU, IPEERL, IDIGIT, IPETY, IRNDOC, II3FLG,  000300  CM*FLAG  2
     1  ENDFLG                                                           000310  CM*FLAG  3
      LOGICAL ENDFLG, ENDREP                                             000320  CM*FLAG  4
      COMMON /PPLCTN/ KPEP, NPEP, KPE, NPE, NVENKE                       000770  CM*PPLCTN2
      COMMON /RUNDAT/ DAT, TIM, NCASE, ICOPEG, IPPEG                     000840  CM*RUNDAT2
      COMMON /TIME/ DURLOS, TIMNOW, TOF, TRP, TIMPA, TIMPAI              001020  CM*TIME  2
C                                                                        002380  PP-COPE 41
      CALL TITLE                                                         002390  PP-COPE 42
      CALL INITLZ                                                        002400  PP-COPE 43
C                                                                        002410  PP-COPE 44
C  BEGIN "CASE" LOOP                                                     002420  PP-COPE 45
C                                                                        002430  PP-COPE 46
  100 CALL INPUT                                                         002440  PP-COPE 47
C                                                                        002450  PP-COPE 48
      CALL PEINTZ                                                        002460  PP-COPE 49
C                                                                        002470  PP-COPE 50
      KPEP=1                                                             002480  PP-COPE 51
      NDOKIL=0                                                           002490  PP-COPE 52

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM

(PAGE 1 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

C                                                                         002500  PP-COPE 53
C  BEGIN "REPLICATION" LOOP                                               002510  PP-COPE 54
C                                                                         002520  PP-COPE 55
  110 KPE=1                                                               002530  PP-COPE 56
      TIMNOW=0.0                                                          002540  PP-COPE 57
      TIMPA=0.0                                                           002550  PP-COPE 58
      ENDREP=.FALSE.                                                      002560  PP-COPE 59
      NVEHKL=0                                                            002570  PP-COPE 60
      DOKILD=.FALSE.                                                      002580  PP-COPE 61
C                                                                         002590  PP-COPE 62
      CALL ARTCHK                                                         002600  PP-COPE 63
      IF (ENDREP) GO TO 140                                               002610  PP-COPE 64
C                                                                         002620  PP-COPE 65
      CALL GETVIS                                                         002630  PP-COPE 66
C                                                                         002640  PP-COPE 67
      CALL GETPNG                                                         002650  PP-COPE 68
C                                                                         002660  PP-COPE 69
      CALL VISCHK                                                         002670  PP-COPE 70
      IF (ENDREP) GO TO 140                                               002680  PP-COPE 71
C                                                                         002690  PP-COPE 72
      CALL PNGCHK                                                         002700  PP-COPE 73
      IF (ENDREP) GO TO 140                                               002710  PP-COPE 74
C                                                                         002720  PP-COPE 75
      CALL DETCTN                                                         002730  PP-COPE 76
      IF (ENDREP) GO TO 140                                               002740  PP-COPE 77
C                                                                         002750  PP-COPE 78
      CALL SMOKE                                                          002760  PP-COPE 79
      IF (ENDREP) GO TO 140                                               002770  PP-COPE 80
C                                                                         002780  PP-COPE 81
      CALL DUST                                                           002790  PP-COPE 82
      IF (ENDREP) GO TO 140                                               002800  PP-COPE 83
C                                                                         002810  PP-COPE 84
      IBAIL=1                                                             002820  PP-COPE 85
      CALL PLNCHK                                                         002830  PP-COPE 86
      IF (ENDREP) GO TO 140                                               002840  PP-COPE 87
C                                                                         002850  PP-COPE 88
      CALL GETTIM                                                         002860  PP-COPE 89
      IF (ENDREP) GO TO 140                                               002870  PP-COPE 90
C                                                                         002880  PP-COPE 91
      IBAIL=2                                                             002890  PP-COPE 92
      CALL BLOCHK                                                         002900  PP-COPE 93
      IF (ENDREP) GO TO 140                                               002910  PP-COPE 94
C                                                                         002920  PP-COPE 95
      IF (IPNDOC.EQ.0) CALL TIMCHK                                        002930  PP-COPE 96
      IF (ENDREP) GO TO 140                                               002940  PP-COPE 97
C                                                                         002950  PP-COPE 98
      CALL ADDTOT                                                         002960  PP-COPE 99
C                                                                         002970  PP-COPE100
C  BEGIN "ROUND" LOOP                                                     002980  PP-COPE101
C                                                                         002990  PP-COPE102
  120 CALL LOSCHK                                                         003000  PP-COPE103
      IF (ENDREP) GO TO 130                                               003010  PP-COPE104
C                                                                         003020  PP-COPE105
      CALL WARNDD                                                         003030  PP-COPE106

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 2 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
      IF (ENDREP) GO TO 130                                                   003040   PP-COPE107
C                                                                             003050   PP-COPE108
      CALL DFCHK                                                              003060   PP-COPE109
      IF (ENDREP) GO TO 130                                                   003070   PP-COPE110
C                                                                             003080   PP-COPE111
      CALL PHDREL                                                             003090   PP-COPE112
      IF (ENDREP) GO TO 130                                                   003100   PP-COPE113
C                                                                             003110   PP-COPE114
      CALL MINTRN                                                             003120   PP-COPE115
      IF (ENDREP) GO TO 130                                                   003130   PP-COPE116
C                                                                             003140   PP-COPE117
      CALL PECHK                                                              003150   PP-COPE118
      IF (ENDREP) GO TO 130                                                   003160   PP-COPE119
C                                                                             003170   PP-COPE120
      CALL HITCHK                                                             003180   PP-COPE121
      IF (ENDREP) GO TO 130                                                   003190   PP-COPE122
C                                                                             003200   PP-COPE123
      CALL PFCHK                                                              003210   PP-COPE124
C                                                                             003220   PP-COPE125
  130 IF (KRF.EQ.NRF) GO TO 140                                               003230   PP-COPE126
C                                                                             003240   PP-COPE127
      CALL ADDTBR                                                             003250   PP-COPE128
C                                                                             003260   PP-COPE129
      KRF=KRF+1                                                               003270   PP-COPE130
      ENDREP=.FALSE.                                                          003280   PP-COPE131
      GO TO 120                                                               003290   PP-COPE132
C                                                                             003300   PP-COPE133
C  END "ROUND" LOOP                                                           003310   PP-COPE134
C                                                                             003320   PP-COPE135
  140 IF (KREP.EQ.NREP) GO TO 150                                             003330   PP-COPE136
      KREP=KREP+1                                                             003340   PP-COPE137
      GO TO 110                                                               003350   PP-COPE138
C                                                                             003360   PP-COPE139
C  END "REPLICATION" LOOP                                                     003370   PP-COPE140
C                                                                             003380   PP-COPE141
  150 CALL OUTPUT                                                             003390   PP-COPE142
C                                                                             003400   PP-COPE143
      IF (.NOT.ENDFLG) GO TO 100                                              003410   PP-COPE144
C                                                                             003420   PP-COPE145
C  END "CASE" LOOP                                                            003430   PP-COPE146
C                                                                             003440   PP-COPE147
      STOP ' IN COPE: NORMAL PROGRAM TERMINATION '                            003450   PP-COPE148
C                                                                             003460   PP-COPE149
      END                                                                     003470   PP-COPE150
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 3 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      BLOCK DATA BDATA1                                                 003490  BD-BDATA12
C                                                                       003500  BD-BDATA13
      COMMON /ABRLBL/ IDABRT(3,19)                                      000130  CM+ABRLBL2
      COMMON /ACHAR/ CHAR(10,120), NREC                                 000150  CM+ACHAR 2
      COMMON /DESRNG/ DESRNG(3,2)                                       000210  CM+DESRNG2
      COMMON /DISPLY/ DISPT(24), DISPM(12), REFLBL(8), TMPLBL(8)        000230  CM+DISPLY2
      COMMON /FLTTIM/ TOFAPY(16)                                        000340  CM+FLTTIM2
      COMMON /FSRESP/ FSRT(14,2)                                        000360  CM+FSRESP2
      COMMON /HEADNG/ ACQLBL(2,4), DAYLBL(2), DESLBL(2), DGTLBL(2,2),   000380  CM+HEADNG2
     1  DOLBL(3,2), POLBL(4,2), PSPLBL(3,2)                             000390  CM+HEADNG3
      COMMON /LOGFLG/ FIRSTL, SEQNML, SPCL(70), SUPTFC, FTSILL          000510  CM+LOGFLG2
      LOGICAL FIRSTL, SEQNML, SPCL, SUPTFC, FTSILL                      000520  CM+LOGFLG3
      COMMON /POINT/ IPOINT(70)                                         000600  CM+POINT 2
      COMMON /SMOKED/ SMK2(3,3,2), SMK5(3,3,2), FRONT, NSMK2, NSMK5,    000920  CM+SMOKED2
     1  PSMKKL                                                          000930  CM+SMOKED3
      COMMON /STITLE/ SLVERS, VERDAT(2)                                 000950  CM+STITLE2
      COMMON /SYMBOL/ BLANK, DECPNT, ALFBET(27), ANUMBR(11), SEP(5),    000970  CM+SYMBOL2
     1  DOLLAR                                                          000980  CM+SYMBOL3
      COMMON /XVALUE/ XVALUE(8,9,2)                                     001100  CM+XVALUE2
C                                                                       003640  BD-BDATA17
C     * * * * *   FILL IN ABRLBL COMMON BLOCK   * * * * *               003650  BD-BDATA18
C                                                                       003660  BD-BDATA19
C                                                                       003670  BD-BDATA20
      DATA ((IDABRT(I,J),I=1,3),J=1,19)/                                003680  BD-BDATA21
     1 10HDESIGNATOR,10H IN INDIRE,10HCT FIRE   ,                       003690  BD-BDATA22
     2 10HTARGET BEY,10HOND VISIBI,10HLITY RANGE,                       003700  BD-BDATA23
     3 10HTARGET BEY,10HOND DESIGN,10HATOR RANGE,                       003710  BD-BDATA24
     4 10HTARGET NOT,10H DETECTED ,10H          ,                       003720  BD-BDATA25
     5 10HSMOKE KILL,10HED MISSION,10H          ,                       003730  BD-BDATA26
     6 10HDUST KILLE,10HD MISSION ,10H          ,                       003740  BD-BDATA27
     7 10HDESIGNATOR,10H BAILED-OU,10HT PRE-COMM,                       003750  BD-BDATA28
     8 10HCOMMO OUT ,10H(BOTH VOIC,10HE & DIGIT),                       003760  BD-BDATA29
     9 10HERROR IN T,10HRANSMISSIO,10HN         ,                       003770  BD-BDATA30
     A 10HDESIGNATOR,10H BAILED-OU,10HT POST-COM,                       003780  BD-BDATA31
     B 10HLOS LOST -,10H NO FIRING,10H          ,                       003790  BD-BDATA32
     C 10HLOS LOST D,10HURING MISS,10HION       ,                       003800  BD-BDATA33
     D 10HDESIGNATOR,10H NOT WARNE,10HD IN TIME ,                       003810  BD-BDATA34
     E 10HDESIGNATOR,10H IN DIRECT,10H FIRE     ,                       003820  BD-BDATA35
     F 10HROUND RELI,10HABILITY FA,10HILURE     ,                       003830  BD-BDATA36
     G 10HTARGET OBS,10HCURED BY M,10HINI-TERRN ,                       003840  BD-BDATA37
     H 10HROUND DID ,10HNOT ENGAGE,10H TARGET   ,                       003850  BD-BDATA38
     I 10HROUND ENGA,10HGED BUT DI,10HD NOT HIT ,                       003860  BD-BDATA39
     J 10HROUND HIT ,10HBUT DID NO,10HT KILL    /                       003870  BD-BDATA40
C                                                                       003880  BD-BDATA41
C                                                                       003890  BD-BDATA42
C     * * * * *   FILL IN ACHAR COMMON BLOCK   * * * * *                003900  BD-BDATA43
C                                                                       003910  BD-BDATA44
      DATA  ((CHAR(I,J),I=1,10),J=1,13)/                                003920  BD-BDATA45
     1 1.,0.,10HFIRSTCASE ,7*1H ,                                       003930  BD-BDATA46
     2 2.,4.,10HNEXTCASE  ,7*1H ,                                       003940  BD-BDATA47
     3 3.,0.,10HENDOFCASE ,7*1H ,                                       003950  BD-BDATA48
     4 3.,0.,10HEND       ,7*1H ,                                       003960  BD-BDATA49
     5 4.,0.,10HENDOFFINAL,10HCASE       ,6*1H ,                        003970  BD-BDATA50
     6 4.,0.,10HENDF      ,7*1H ,                                       003980  BD-BDATA51

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 4 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
     7 5.,4.,10HPLOT       ,7*1H ,                                                004000  BD-BDATA52
     8 6.,0.,10HOVERRIDE   ,7*1H ,                                                004000  BD-BDATA53
     9 6.,0.,10HOVERIDE    ,7*1H ,                                                004010  BD-BDATA54
     A 7.,0.,10HTEMPORARY ,7*1H ,                                                 004020  BD-BDATA55
     B 7.,0.,10HTEMP       ,7*1H ,                                                004030  BD-BDATA56
     C 8.,0.,10HRESET      ,7*1H ,                                                004040  BD-BDATA57
     D 9.,3.,10HCONTROL    ,7*1H /                                                004050  BD-BDATA58
C                                                                                 004060  BD-BDATA59
      DATA  ((CHAR(I,J),I=1,10),J=14,23)/                                         004070  BD-BDATA60
     1 11.,4.,10HLINEOFDESIGN,10HTMODE         ,6*1H ,                            004080  BD-BDATA61
     2 11.,4.,10HLOSMODE    ,7*1H ,                                               004090  BD-BDATA62
     3 12.,6.,10HMISSIONCOD,10HE              ,6*1H ,                             004100  BD-BDATA63
     4 12.,6.,10HMSNCODE    ,7*1H ,                                               004110  BD-BDATA64
     5 13.,4.,10HNUMBEROFDI,10HFFERENTTAR,10HGETPOSTURE,1HS,4*1H ,                004120  BD-BDATA65
     6 13.,4.,10HNUMDIFTGTP,7*1H ,                                                004130  BD-BDATA66
     7 14.,7.,10HTARGETTYPE,7*1H ,                                                004140  BD-BDATA67
     8 16.,6.,10HDESIGNATOR,10HTYPE            ,6*1H ,                            004150  BD-BDATA68
     9 16.,6.,10HDESIGTYPE ,7*1H ,                                                004160  BD-BDATA69
     A 17.,4.,10HDOPOSITION,7*1H /                                                004170  BD-BDATA70
C                                                                                 004180  BD-BDATA71
      DATA  ((CHAR(I,J),I=1,10),J=24,37)/                                         004190  BD-BDATA72
     1 21.,10.,10HWEATHER    ,7*1H ,                                              004200  BD-BDATA73
     2 21.,10.,10HW          ,7*1H ,                                              004210  BD-BDATA74
     3 22.,8.,10HACQUISITIO,10HNRANGEDIST,10HRIBUTION    ,5*1H ,                  004220  BD-BDATA75
     4 22.,8.,10HACQRNGDIST,7*1H ,                                                004230  BD-BDATA76
     5 23.,12.,10HRESPONSETI,10HME             ,6*1H ,                            004240  BD-BDATA77
     6 23.,12.,10HRESPTIME   ,7*1H ,                                              004250  BD-BDATA78
     7 24.,4.,10HDIRECTFIRE,10HSUPPRESSIO,10HN              ,5*1H ,               004260  BD-BDATA79
     8 24.,4.,10HDFIRESUPPR,7*1H ,                                                004270  BD-BDATA80
     9 25.,0.,10HRANDOMOCCU,10HRENCEDIST,10HRIBUTION   ,5*1H ,                   004280  BD-BDATA81
     A 25.,0.,10HRODIST      ,7*1H ,                                              004290  BD-BDATA82
     B 26.,0.,10HREDATA      ,7*1H ,                                              004300  BD-BDATA83
     C 27.,4.,10HTARGETPOST,10HUREDISTRIB,10HUTION          ,5*1H ,               004310  BD-BDATA84
     D 27.,4.,10HTGTPSTDIST,7*1H ,                                                004320  BD-BDATA85
     E 28.,0.,10HINVARIANT ,7*1H /                                                004330  BD-BDATA86
C                                                                                 004340  BD-BDATA87
      DATA  ((CHAR(I,J),I=1,10),J=38,39)/                                         004350  BD-BDATA88
     1 31.,0.,10HRUMC        ,7*1H ,                                              004360  BD-BDATA89
     2 32.,4.,10HSMOKE       ,7*1H /                                              004370  BD-BDATA90
C                                                                                 004380  BD-BDATA91
      DATA  ((CHAR(I,J),I=1,10),J=40,55)/                                         004390  BD-BDATA92
     1 41.,0.,10HDOARTILLER,10HYPK           ,6*1H ,                              004400  BD-BDATA93
     2 41.,0.,10HDOAPPK      ,7*1H ,                                              004410  BD-BDATA94
     3 42.,0.,10HTARGETVELO,10HCITY          ,6*1H ,                              004420  BD-BDATA95
     4 42.,0.,10HTGTVEL      ,7*1H ,                                              004430  BD-BDATA96
     5 43.,0.,10HMAXIMUMDES,10HIGNATORRAN,10HGE             ,5*1H ,               004440  BD-BDATA97
     6 43.,0.,10HMAXDESRNG ,7*1H ,                                                004450  BD-BDATA98
     7 44.,0.,10HBAILOUTRAN,10HGE            ,6*1H ,                              004460  BD-BDATA99
     8 44.,0.,10HBAILOUTRNG,7*1H ,                                                004470  BD-BDAT100
     9 45.,0.,10HTIMEOFFLIG,10HHT            ,6*1H ,                              004480  BD-BDAT101
     A 45.,0.,10HTOF         ,7*1H ,                                              004490  BD-BDAT102
     B 46.,0.,10HANGLET      ,7*1H ,                                              004500  BD-BDAT103
     C 47.,0.,10HDUST        ,7*1H ,                                              004510  BD-BDAT104
     D 48.,0.,10HPROBABILIT,10HYOFSUCCESS,10HFUL?OICETP,10HANDMISSION,4*004520    BD-BDAT105
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1  FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 5 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      EIH ,                                                               004530  BD-BDAT106
      F 48.,0.,10HPREVOCTRAN,7*1H ,                                       004540  BD-BDAT107
      G 49.,0.,10HPROBABILIT,10HYOFSUCCESS,10HFULDIGITAL,10HTRANSMISSI,10004550  BD-BDAT108
      HHON         ,3*1H ,                                                004560  BD-BDAT109
      I 49.,0.,10HPRBDGTTRAN,7*1H /                                       004570  BD-BDAT110
C                                                                         004580  BD-BDAT111
       DATA  ((CHAR(I,J),I=1,10),J=56,73)/                                004590  BD-BDAT112
      1 50.,0.,10HROUNDINFLI,10HGHTRELIABI,10HLITY       ,5*1H ,          004600  BD-BDAT113
      2 50.,0.,10HRNDFLTREL ,7*1H ,                                       004610  BD-BDAT114
      3 51.,0.,10HTIMEBETWEE,10HNROUNDS    ,6*1H ,                        004620  BD-BDAT115
      4 51.,0.,10HTBR        ,7*1H ,                                      004630  BD-BDAT116
      5 52.,0.,10HNUMBEROFRE,10HPLICATIONS,6*1H ,                         004640  BD-BDAT117
      6 52.,0.,10HNUMREP     ,7*1H ,                                      004650  BD-BDAT118
      7 53.,0.,10HNUMBEROFRO,10HUNDSTOREFI,10HRED        ,5*1H ,          004660  BD-BDAT119
      8 53.,0.,10HNUMRNDS    ,7*1H ,                                      004670  BD-BDAT120
      9 54.,0.,10HNUMBEROFVE,10HHICLESPERT,10HARGET      ,5*1H ,          004680  BD-BDAT121
      A 54.,0.,10HNUMVEHTGT ,7*1H ,                                       004690  BD-BDAT122
      B 55.,0.,10HDISTANCEBE,10HTWEENVEHIC,10HLES        ,5*1H ,          004700  BD-BDAT123
      C 55.,0.,10HDISTBVEH  ,7*1H ,                                       004710  BD-BDAT124
      D 56.,0.,10HGUNTARGETR,10HANGE       ,6*1H ,                        004720  BD-BDAT125
      E 56.,0.,10HGTRNG      ,7*1H ,                                      004730  BD-BDAT126
      F 57.,0.,10HREFLECTIVI,10HTY         ,6*1H ,                        004740  BD-BDAT127
      G 57.,0.,10HREFLECT    ,7*1H ,                                      004750  BD-BDAT128
      H 58.,0.,10HDEFLECTION,10HBIAS       ,6*1H ,                        004760  BD-BDAT129
      I 58.,0.,10HDEFLB      ,7*1H /                                      004770  BD-BDAT130
C                                                                         004780  BD-BDAT131
       DATA  ((CHAR(I,J),I=1,10),J=74,81)/                                004790  BD-BDAT132
      1 59.,0.,10HTARGETHEAD,10HING        ,6*1H ,                        004800  BD-BDAT133
      2 59.,0.,10HTGTHDG     ,7*1H ,                                      004810  BD-BDAT134
      3 60.,0.,10HSEEKERSENS,10HITIVITY    ,6*1H ,                        004820  BD-BDAT135
      4 60.,0.,10HSEEKSENS   ,7*1H ,                                      004830  BD-BDAT136
      5 61.,0.,10HPROBABILIT,10HYOFACORREC,10HTMESSAGE   ,5*1H ,          004840  BD-BDAT137
      6 61.,0.,10HPROBCORMSG,7*1H ,                                       004850  BD-BDAT138
      7 62.,0.,10HPROBABILIT,10HYDOWARNEDT,10HOLASE      ,5*1H ,          004860  BD-BDAT139
      8 62.,0.,10HPROBDOWARN,7*1H /                                       004870  BD-BDAT140
C                                                                         004880  BD-BDAT141
       DATA NPEC /81/                                                     004890  BD-BDAT142
C                                                                         004900  BD-BDAT143
C                                                                         004910  BD-BDAT144
C      * * * * *   FILL IN DESPNG COMMON BLOCK   * * * * *                004920  BD-BDAT145
C                                                                         004930  BD-BDAT146
       DATA ((DESPNG(I,J),I=1,3),J=1,2) /7000., 7000., 3500., 2600., 2600004940  BD-BDAT147
      1., 2600./                                                          004950  BD-BDAT148
C                                                                         004960  BD-BDAT149
C      * * * * *   FILL IN DISPLY COMMON BLOCK   * * * * *                004970  BD-BDAT150
C                                                                         004980  BD-BDAT151
       DATA (DISPT(I),I=1,3) /4H1400, 4H0600, 4H2200/                     004990  BD-BDAT152
       DATA (DISPM(I),I=1,4) /3HJUN, 3HDEC, 3HMAR, 3HSEP/                 005000  BD-BDAT153
       DATA (RECLBL(I),I=1,8) /1H , 6HACODAT, 6HESPDAT, 6HDESDAT, 5HRODAV005010  BD-BDAT154
      1, 1H , 6HRSTDAT, 6HAINVDA/                                         005020  BD-BDAT155
       DATA (TMPLBL(I),I=1,8) /5HWDATA, 6HACODAT, 6HESPDAT, 6HDESDAT, 6HR005030  BD-BDAT156
      1ODATA, 6HREDATA, 6HRSTDAT, 6HAINVLA/                               005040  BD-BDAT157
C                                                                         005050  BD-BDAT158
C      * * * * *   FILL IN FLTTIM COMMON BLOCK   * * * * *                005060  BD-BDAT159

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 6 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

C                                                                              005070   BD-BDAT160
      DATA (TOFARY(I),I=1,16) /7*0., 31., 3*0., 44.4, 3*0., 67./               005080   BD-BDAT161
C                                                                              005090   BD-BDAT162
C     * * * * *   FILL IN FCPESP COMMON BLOCK   * * * * *                      005100   BD-BDAT163
C                                                                              005110   BD-BDAT164
      DATA ((FSPT(I,J),I=1,14),J=1,2) /.0, .03, .21, .60, .75, .83, .88,005120   BD-BDAT165
     1 .91, .93, .95, .97, .98, .99, 1.0, 20., 40., 60., 80., 100., 120.005130   BD-BDAT166
     2, 140., 160., 180., 200., 220., 240., 260., 280./                        005140   BD-BDAT167
C                                                                              005150   BD-BDAT168
C     * * * * *   FILL IN HEADNG COMMON BLOCK   * * * * *                      005160   BD-BDAT169
C                                                                              005170   BD-BDAT170
      DATA ((ACQLBL(I,J),I=1,2),J=1,4) /10HAVERAGE TE, 5HRRAIN, 10HCLOSE005180   BD-BDAT171
     1 TERR, 3HAIN, 10HOPEN TERRA, 2HIN, 10HSTAGS TERR, 3HAIN/                 005190   BD-BDAT172
      DATA (DAYLBL(I),I=1,2) /8HDAY TIME, 10HNIGHT TIME/                       005200   BD-BDAT173
      DATA (DFSLBL(I),I=1,2) /4HHIGH, 4HNONE/                                  005210   BD-BDAT174
      DATA ((DGTLBL(I,J),I=1,2),J=1,2) /10HDIGITAL CO, 3HMMO, 10HVOICE C005220   BD-BDAT175
     10HM, 1HO/                                                                005230   BD-BDAT176
      DATA ((DOLBL(I,J),I=1,3),J=1,2) /10HDESIGNATOR, 10H AT VANTAG, 7HE005240   BD-BDAT177
     1 POINT, 10HDESIGNATOR, 10H WITH MANE, 10HUVER UNIT /                     005250   BD-BDAT178
      DATA ((ROLBL(I,J),I=1,4),J=1,2) /10HCONTINUOUS, 10H LOS ("SHO, 10H005260   BD-BDAT179
     1OTING GALL, 5HERY"), 10HBROKEN LOS, 10H ("RANDOM , 10HOCCURRENCE, 005270   BD-BDAT180
     22H")/                                                                    005280   BD-BDAT181
      DATA ((RSPLBL(I,J),I=1,3),J=1,3) /10HPRE-PLANNE, 10HD TARGET  , 1H005290   BD-BDAT182
     1 , 10HTARGET OF , 10HOPPORTUNIT, 1HY, 10HPRIORITY P, 10HRE-PLANNED005300   BD-BDAT183
     2, 7H TARGET/                                                             005310   BD-BDAT184
C                                                                              005320   BD-BDAT185
C     * * * * *   FILL IN LOGFLG COMMON BLOCK   * * * * *                      001420   DT+LOGFLG2
C                                                                              001430   DT+LOGFLG3
      DATA FIRSTL /.TRUE./ , SEONML /.FALSE./                                  001440   DT+LOGFLG4
      DATA (SPCL(I),I=1,70) /70*.FALSE./                                       001450   DT+LOGFLG5
      DATA SHRTFC /.TRUE./                                                     001460   DT+LOGFLG6
C                                                                              005340   BD-BDAT187
C     * * * * *   FILL IN POINT COMMON BLOCK   * * * * *                       005350   BD-BDAT188
C                                                                              005360   BD-BDAT189
      DATA (IPOINT(I),I=1,70) /1, 2, 3, 4, 66*5/                               005370   BD-BDAT190
C                                                                              005380   BD-BDAT191
C     * * * * *   FILL IN (PART OF) SMOKED COMMON BLOCK   * * * * *            005390   BD-BDAT192
C                                                                              005400   BD-BDAT193
      DATA (((SMK2(I,J,K),I=1,3),J=1,3),K=1,2) /1936., 376., 220., 0., 8005410   BD-BDAT194
     176., 0., 0., 1220., 0., 1260., 280., 178., 0., 530., 0., 0., 826.,005420   BD-BDAT195
     2 0./                                                                     005430   BD-BDAT196
      DATA (((SMK5(I,J,K),I=1,3),J=1,3),K=1,2) /1230., 240., 168., 0., 5005440   BD-BDAT197
     116., 0., 0., 800., 0., 656., 160., 118., 0., 300., 0., 0., 436., 0005450   BD-BDAT198
     2./                                                                       005460   BD-BDAT199
      DATA FRONT /12.0/                                                        005470   BD-BDAT200
C                                                                              005480   BD-BDAT201
C     * * * * *   FILL IN STITLE COMMON BLOCK   * * * * *                      005490   BD-BDAT202
C                                                                              005500   BD-BDAT203
      DATA SLVERS /6.2/ , (VERDAT(I),I=1,2) /10H6 MARCH 19, 2H80/              005510   BD-BDAT204
C                                                                              005520   BD-BDAT205
C                                                                              001480   DT+SYMBOL2
C     * * * * *   FILL IN SYMBOL COMMON BLOCK   * * * * *                      001490   DT+SYMBOL3
C                                                                              001500   DT+SYMBOL4
      DATA BLANK /1H / , PERCNT /1H./                                          001510   DT+SYMBOL5

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 7 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      DATA ALFPET /1HA, 1HB, 1HC, 1HD, 1HE, 1HF, 1HG, 1HH, 1HI, 1HJ, 1HK001520  DT+SYMBOL6
     1, 1HL, 1HM, 1HN, 1HO, 1HP, 1HQ, 1HR, 1HS, 1HT, 1HU, 1HV, 1HW, 1HX,001530  DT+SYMBOL7
     2 1HY, 1HZ, 1H*/                                                  001540  DT+SYMBOL8
      DATA ANUMBR /1H0, 1H1, 1H2, 1H3, 1H4, 1H5, 1H6, 1H7, 1H8, 1H9, 1H-001550  DT+SYMBOL9
     1/                                                                001560  DT+SYMBO10
      DATA SEP /1H,, 1H/, 1H?, 1H(, 1H)/                               001570  DT+SYMBO11
      DATA DOLLAR /1H$/                                                001580  DT+SYMBO12
C                                                                      001590  DT+SYMBO13
C                                                                      005540  BD-BDAT207
C                                                                      001610  DT+XVALUE2
C     * * * * *   FILL IN XVALUE COMMON BLOCK    * * * * *             001620  DT+XVALUE3
C                                                                      001630  DT+XVALUE4
      DATA (((XVALUE(I,J,K),I=1,8),K=1,2),J=1,9) /                     001640  DT+XVALUE5
C          DUMMY SLOTS FOR NOMINAL RESPONSE TIMES                      001650  DT+XVALUE6
     1      8*0.,                                                      001660  DT+XVALUE7
     2      8*8.,                                                      001670  DT+XVALUE8
C          DESIGNATOR TYPES                                            001680  DT+XVALUE9
     1      1.,     2.,     3.,     5*0.,                              001690  DT+XVALU10
     2      0.,     1.,     1.,     5*8.,                              001700  DT+XVALU11
C          TARGET VELOCITIES                                           001710  DT+XVALU12
     1      2.,     3.,     5.,     8.,     9.,     3*0.,              001720  DT+XVALU13
     2      0.,     0.,     1.,     2.,     2.,     3*8.,              001730  DT+XVALU14
C          GUN-TARGET RANGES                                           001740  DT+XVALU15
     1      8.,     12.,    20.,    30.,    40.,    10.,    16.,    0.,001750  DT+XVALU16
     2      0.,     1.,     2.,     3.,     4.,     5.,     6.,     8.,001760  DT+XVALU17
C          REFLECTIVITIES                                              001770  DT+XVALU18
     1      .05,    .10,    .20,    .30,    4*0.,                      001780  DT+XVALU19
     2      0.,     1.,     2.,     3.,     4*8.,                      001790  DT+XVALU20
C          ANGLES T                                                    001800  DT+XVALU21
     1      0.,     25.,    30.,    60.,    90.,    120.,   2*0.,      001810  DT+XVALU22
     2      0.,     1.,     2.,     3.,     4.,     5.,     2*8.,      001820  DT+XVALU23
C          DEFLECTION BIASES                                           001830  DT+XVALU24
     1      -200., -100., 0.,     100.,   200.,   3*0.,                001840  DT+XVALU25
     2      0.,     1.,     2.,     3.,     4.,     3*8.,              001850  DT+XVALU26
C          TARGET HEADINGS                                             001860  DT+XVALU27
     1      -60.,   -30.,   0.,     30.,    60.,    3*0.,              001870  DT+XVALU28
     2      0.,     1.,     2.,     3.,     4.,     3*8.,              001880  DT+XVALU29
C          SEEKER SENSITIVITIES                                        001890  DT+XVALU30
CSFC  THE FOLLOWING CARD CONTAINS FALSE SEEKER SENSITIVITY VALUES TO   001900  DT+XVALU31
CSFC  AVOID INCLUDING CLASSIFIED NUMBERS IN THE PROGRAM LISTING.       001910  DT+XVALU32
CSFC  THE CORRECT VALUES SHOULD BE RESTORED WHEN RUNNING IN A SECURE   001920  DT+XVALU33
CSFC  ENVIRONMENT.  RUNS DONE WITH THESE FALSE VALUES ARE FOR EXAMPLE  001930  DT+XVALU34
CSFC  PURPOSES ONLY.                                                   001940  DT+XVALU35
     1      24.,    36.,    48.,    60.,    72.,    3*0.,              001950  DT+XVALU36
     2      0.,     1.,     2.,     3.,     4.,     3*8./              001960  DT+XVALU37
C                                                                      001970  DT+XVALU38
      END                                                              005560  BD-BDAT209

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1  FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 8 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      FUNCTION FPRCNT (XIN,PAREN)                                            005580  FN-FPRCNT2
C                                                                            005590  FN-FPRCNT3
      LOGICAL PAREN                                                          005600  FN-FPRCNT4
C                                                                            005610  FN-FPRCNT5
      IF (XIN.GE.100.05) STOP ' IN FPRCNT: ERROR NUMBER 1 '                  005620  FN-FPRCNT6
      IF (.NOT.PAREN) GO TO 100                                              005630  FN-FPRCNT7
C                                                                            005640  FN-FPRCNT8
      IF (XIN.LT.0.0) XOUT=6HUNDEF)                                          005650  FN-FPRCNT9
      IF (0.0.LE.XIN.AND.XIN.LT.1.) ENCODE (6,120,XOUT) XIN                  005660  FN-FPRCN10
      IF (1..LE.XIN.AND.XIN.LT.10.) ENCODE (7,130,XOUT) XIN                  005670  FN-FPRCN11
      IF (10.LE.XIN.AND.XIN.LT.100.) ENCODE (7,140,XOUT) XIN                 005680  FN-FPRCN12
      IF (XIN.GE.100.) XOUT=7H100.0%)                                        005690  FN-FPRCN13
      FPRCNT=XOUT                                                            005700  FN-FPRCN14
      GO TO 110                                                              005710  FN-FPRCN15
C                                                                            005720  FN-FPRCN16
  100 IF (XIN.LT.0.) XOUT=10H     UNDEF                                      005730  FN-FPRCN17
      IF (0.0.LE.XIN.AND.XIN.LT.100.) ENCODE (10,150,XOUT) XIN               005740  FN-FPRCN18
      IF (XIN.GE.100.) XOUT=10H    100.0%                                    005750  FN-FPRCN19
      FPRCNT=XOUT                                                            005760  FN-FPRCN20
C                                                                            005770  FN-FPRCN21
  110 RETURN                                                                 005780  FN-FPRCN22
C                                                                            005790  FN-FPRCN23
C                                                                            005800  FN-FPRCN24
C  * * *  F O R M A T   S T A T E M E N T S  * * *                           005810  FN-FPRCN25
C                                                                            005820  FN-FPRCN26
  120 FORMAT (1H0,F3.2,2H%))                                                 005830  FN-FPRCN27
  130 FORMAT (F4.2,2H%))                                                     005840  FN-FPRCN28
  140 FORMAT (F5.2,2H%))                                                     005850  FN-FPRCN29
  150 FORMAT (4X,F5.2,1H%)                                                   005860  FN-FPRCN30
      END                                                                    005870  FN-FPRCN31
C                                                                            005880  FN-FPRCN32

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 9 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      FUNCTION GAMMA (ALPHA,IPN)                                                005900   FN-GAMMA 2
      DIMENSION WK(6), P(1)                                                     005910   FN-GAMMA 3
      IN=1                                                                      005920   FN-GAMMA 4
      CALL GGAMA (IPN,ALPHA,IN,WK,P)                                            005930   FN-GAMMA 5
      GAMMA=P(1)                                                                005940   FN-GAMMA 6
      RETURN                                                                    005950   FN-GAMMA 7
      END                                                                       005960   FN-GAMMA 8

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1  FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 10 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)

      FUNCTION IDCHAR (A,N,I)                                          005980   FN-IDCHAR2
C                                                                      005990   FN-IDCHAR3
      COMMON /SYMBOL/ BLANK, DECPNT, ALFBET(27), ANUMBR(11), SEP(5),   000970   CM*SYMBOL2
     1  DOLLAR                                                         000980   CM*SYMBOL3
      DIMENSION A(N)                                                   006010   FN-IDCHAR5
C                                                                      006020   FN-IDCHAR6
      IF (A(I).NE.BLANK) GO TO 100                                     006030   FN-IDCHAR7
      IDCHAR=3                                                         006040   FN-IDCHAR8
      RETURN                                                           006050   FN-IDCHAR9
  100 IF (A(I).NE.DECPNT) GO TO 110                                    006060   FN-IDCHA10
      IDCHAR=2                                                         006070   FN-IDCHA11
      RETURN                                                           006080   FN-IDCHA12
  110 DO 120 J=1,11                                                    006090   FN-IDCHA13
      IF (A(I).NE.ANUMBR(J)) GO TO 120                                 006100   FN-IDCHA14
      IDCHAR=0                                                         006110   FN-IDCHA15
      RETURN                                                           006120   FN-IDCHA16
  120 CONTINUE                                                         006130   FN-IDCHA17
      DO 130 J=1,5                                                     006140   FN-IDCHA18
      IF (A(I).NE.SEP(J)) GO TO 130                                    006150   FN-IDCHA19
      IDCHAR=4                                                         006160   FN-IDCHA20
      RETURN                                                           006170   FN-IDCHA21
  130 CONTINUE                                                         006180   FN-IDCHA22
      DO 140 J=1,27                                                    006190   FN-IDCHA23
      IF (A(I).NE.ALFBET(J)) GO TO 140                                 006200   FN-IDCHA24
      IDCHAR=1                                                         006210   FN-IDCHA25
      RETURN                                                           006220   FN-IDCHA26
  140 CONTINUE                                                         006230   FN-IDCHA27
      IF (A(I).NE.DOLLAR) GO TO 150                                    006240   FN-IDCHA28
      IDCHAR=5                                                         006250   FN-IDCHA29
      RETURN                                                           006260   FN-IDCHA30
  150 IDCHAR=6                                                         006270   FN-IDCHA31
      RETURN                                                           006280   FN-IDCHA32
      END                                                              006290   FN-IDCHA33

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 11 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      LOGICAL FUNCTION NUMRIC(B,N1,N2,IB)                                   006310  FN-NUMRIC2
C                                                                           006320  FN-NUMRIC3
      COMMON /SYMBOL/ BLANK, DECPNT, ALFBET(27), ANUMBR(11), SEP(5),        000970  CM+SYMBOL2
     1 DOLLAR                                                               000980  CM+SYMBOL3
      DIMENSION B(N1,N2), C(80)                                             006340  FN-NUMRIC5
C                                                                           006350  FN-NUMRIC6
      NUMRIC=.FALSE.                                                        006360  FN-NUMRIC7
      DECFLG=0.0                                                            006370  FN-NUMRIC8
      DECODE (80,160,B(1,IB) )(C(IC),IC=1,80)                               006380  FN-NUMRIC9
C                                                                           006390  FN-NUMRI10
      KI=1                                                                  006400  FN-NUMRI11
  100 IF (KI.GT.80) GO TO 140                                               006410  FN-NUMRI12
      IDCHR=IDCHAR(C,80,KI)                                                 006420  FN-NUMRI13
      IF (IDCHR.NE.0) GO TO 110                                             006430  FN-NUMRI14
      KI=KI+1                                                               006440  FN-NUMRI15
      GO TO 100                                                             006450  FN-NUMRI16
  110 IF (IDCHR.NE.2) GO TO 130                                             006460  FN-NUMRI17
      IF (DECFLG.EQ.0.0) GO TO 120                                          006470  FN-NUMRI18
      WRITE (6,170) IB,((B(I1,I2),I1=1,N1),I2=1,N2)                         006480  FN-NUMRI19
      STOP ' IN NUMRIC: ERROR NUMBER 1 '                                    006490  FN-NUMRI20
  120 DECFLG=1.0                                                            006500  FN-NUMRI21
      KI=KI+1                                                               006510  FN-NUMRI22
      GO TO 100                                                             006520  FN-NUMRI23
  130 IF (IDCHR.NE.3) RETURN                                                006530  FN-NUMRI24
  140 IF (DECFLG.NE.0.0) GO TO 150                                          006540  FN-NUMRI25
      C(KI)=DECPNT                                                          006550  FN-NUMRI26
      ENCODE (80,180,B(1,IB) )(C(I),I=1,80)                                 006560  FN-NUMRI27
  150 NUMRIC=.TRUE.                                                         006570  FN-NUMRI28
      IF (KI.LE.10) RETURN                                                  006580  FN-NUMRI29
      WRITE (6,190) IB,((B(I1,IB),I1=1,N1),I2=1,N2)                         006590  FN-NUMRI30
      STOP ' IN NUMRIC: ERROR NUMBER 2 '                                    006600  FN-NUMRI31
C                                                                           006610  FN-NUMRI32
C                                                                           006620  FN-NUMRI33
C   * * *  F O R M A T   S T A T E M E N T S  * * *                         006630  FN-NUMRI34
C                                                                           006640  FN-NUMRI35
  160 FORMAT (80A1)                                                         006650  FN-NUMRI36
  170 FORMAT (1H ,17H IN NUMRIC.  LINE,I3,30H HAD TOO MANY DECIMAL POINT006660  FN-NUMRI37
     1S :,/,(1X,8A10))                                                      006670  FN-NUMRI38
  180 FORMAT (80A1)                                                         006680  FN-NUMRI39
  190 FORMAT (1H ,17H IN NUMRIC.  LINE,I3,49H EXCEEDS ALLOWABLE FIELD WI006690  FN-NUMRI40
     1DTH FOR NUMERIC DATA :,/,(1X,8A10))                                   006700  FN-NUMRI41
      END                                                                   006710  FN-NUMRI42

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 12 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
      FUNCTION URAN31 (I)                                                 006730  FN-URAN312
      IF (I) 110,100,110                                                  006740  FN-URAN313
  100 I=11111111                                                          006750  FN-URAN314
  110 J=I                                                                 006760  FN-URAN315
      J=J*25                                                              006770  FN-URAN316
      J=J-(J/67108864)*67108864                                           006780  FN-URAN317
      J=J*25                                                              006790  FN-URAN318
      J=J-(J/67108864)*67108864                                           006800  FN-URAN319
      J=J*5                                                               006810  FN-URAN310
      J=J-(J/67108864)*67108864                                           006820  FN-URAN311
      A1=J                                                                006830  FN-URAN312
      I=J                                                                 006840  FN-URAN313
      URAN31=A1/67108864.                                                 006850  FN-URAN314
      RETURN                                                              006860  FN-URAN315
      END                                                                 006870  FN-URAN316
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 13 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE ABRTTL (IABRT)                                         006890  SR-ABRTTL2
C                                                                       006900  SP-ABRTTL3
      COMMON /ABORT/ NABORT(20,6), MSAPLM                               000110  CM+ABORT 2
      COMMON /FLAG/ ENDPEP, IDN, IPPEPL, IDIGTL, IPPTY, IRNDOC, IT3FLG, 000300  CM+FLAG  2
     1  ENDFLG                                                          000310  CM+FLAG  3
      LOGICAL ENDFLG, ENDPEP                                            000320  CM+FLAG  4
      COMMON /RPLCTN/ KPEP, NPEP, KPF, NPF, NVEHKL                      000770  CM+RPLCTN2
C                                                                       006940  SP-ABRTTL7
      IPF=1                                                             006950  SP-ABRTTL8
      IF (IABRT.LE.MSAPLM) GO TO 100                                    006960  SP-ABRTTL9
      NABORT(IABRT,KPF)=NABORT(IABRT,KPF)+1                             006970  SP-ABRTT10
      GO TO 120                                                         006980  SP-ABRTT11
C                                                                       006990  SP-ABRTT12
  100 DO 110 IJ=IPF,NPF                                                 007000  SP-ABRTT13
      NABORT(IABRT,IJ)=NABORT(IABRT,IJ)+1                               007010  SP-ABRTT14
  110 CONTINUE                                                          007020  SP-ABRTT15
C                                                                       007030  SP-ABRTT16
  120 ENDPEP=.TRUE.                                                     007040  SP-ABRTT17
C                                                                       007050  SP-ABRTT18
      RETURN                                                            007060  SP-ABRTT19
C                                                                       007070  SP-ABRTT20
      END                                                               007080  SP-ABRTT21

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 14 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      SUBROUTINE ADDTBR                                                 007100  SP-ADDTBR2
C                                                                       007110  SP-ADDTBR3
      COMMON /TIME/ DUPLOS, TIMNOW, TOF, TBR, TIMPA, TIMPA1             001020  CM+TIME  2
C                                                                       007130  SP-ADDTBR5
      TIMPA=TIMPA+TBR                                                   007140  SP-ADDTBR6
C                                                                       007150  SP-ADDTBR7
      RETURN                                                            007160  SP-ADDTBR8
C                                                                       007170  SP-ADDTBR9
      END                                                               007180  SP-ADDTBR10

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 15 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE ADDTOF                                                 007200  SP-ADDTOF2
C                                                                       007210  SP-ADDTOF3
      COMMON /TIME/ DUPLOS, TIMNOW, TOF, TBF, TIMPA, TIMPA1             001020  CM+TIME  2
C                                                                       007230  SP-ADDTOF5
      TIMPA=TIMNOW+TOF                                                  007240  SP-ADDTOF6
C                                                                       007250  SP-ADDTOF7
      RETURN                                                            007260  SP-ADDTOF8
C                                                                       007270  SP-ADDTOF9
      END                                                               007280  SP-ADDTO10

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 16 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE ARTCHK                                                 007300  SP-ARTCHK2
C                                                                       007310  SP-ARTCHK3
      COMMON /KTEST/ KTEST(20,6)                                        000490  CM+KTEST 2
      COMMON /MISC/ DOARPK, POSTKL, PELIF                               000540  CM+MISC  2
      COMMON /RANDOM/ IP1, IP2, IP3, IP4, IP5, IP6, IP7, IP8, IP9, IP10,000640  CM+RANDOM2
     1  IP11, IP12, IP13, IP14, IP15, IP16, IP17, IP18, IP19, IP20, IP21000650  CM+RANDOM3
     2  , IP22, IP23, IP24, IP25, IP26, IP27, IP28                      000660  CM+RANDOM4
      COMMON /RPLCTN/ KPEP, NPEP, KPF, NPF, NVEHKL                      000770  CM+RPLCTN2
C                                                                       007360  SP-ARTCHK8
      RN1=URAN31(IP1)                                                   007370  SP-ARTCHK9
C                                                                       007380  SP-ARTCH10
      KTEST(1,KPF)=KTEST(1,KPF)+1                                       007390  SP-ARTCH11
      IF (RN1.GT.DOARPK) GO TO 100                                      007400  SP-ARTCH12
C                                                                       007410  SP-ARTCH13
      IABRT=1                                                           007420  SP-ARTCH14
      CALL ABRTIL (IABRT)                                               007430  SP-ARTCH15
C                                                                       007440  SP-ARTCH16
  100 RETURN                                                            007450  SP-ARTCH17
C                                                                       007460  SP-ARTCH18
      END                                                               007470  SP-ARTCH19

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 17 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)

      SUBROUTINE BLOCHK                                                 007490  SP-BLOCHK2
C                                                                       007500  SP-BLOCHK3
      COMMON /BAIL/ BLORNG, IBAIL                                       000170  CM+BAIL  2
      COMMON /KTEST/ KTEST(20,6)                                        000490  CM+KTEST 2
      COMMON /RANGE/ DTRNG, MXDRNG, VISLIM, RNGNOW                      000680  CM+RANGE 2
      REAL MXDRNG                                                       000690  CM+RANGE 3
      COMMON /RPLCTN/ KREP, NREP, KPF, NPF, NVEHKL                      000770  CM+RPLCTN2
      COMMON /TARGET/ TVEL, NVEHCL, DISPVH, ITGTTP, IMUTS, ITGTPS       001000  CM+TARGET2
      COMMON /TIME/ DUELOS, TIMNOW, TOF, TRP, TIMFA, TIMPA1             001020  CM+TIME  2
C                                                                       007570  SP-BLOCH10
      IF (IBAIL.EQ.1) KTEST(7,KPF)=KTEST(7,KPF)+1                       007580  SP-BLOCH11
      IF (IBAIL.EQ.2) KTEST(10,KPF)=KTEST(10,KPF)+1                     007590  SP-BLOCH12
      IF ((DTRNG-TVEL*TIMNOW).GT.BLORNG) GO TO 100                      007600  SP-BLOCH13
C                                                                       007610  SP-BLOCH14
      IABRT=7                                                           007620  SR-BLOCH15
      IF (IBAIL.EQ.2) IABRT=10                                          007630  SP-BLOCH16
      CALL ABRTTL (IABRT)                                               007640  SP-BLOCH17
C                                                                       007650  SP-BLOCH18
  100 RETURN                                                            007660  SP-BLOCH19
C                                                                       007670  SP-BLOCH20
      END                                                               007680  SP-BLOCH21

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 18 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE CCPLOT                                                   007700  SP-CCPLOT2
C                                                                         007710  SP-CCPLOT3
      COMMON /RUNDAT/ DAT, TIM, NCASE, ICCPFG, IPPFG                      000840  CM+RUNDAT2
C                                                                         007730  SP-CCPLOT5
      DIMENSION CM(10)                                                    007740  SP-CCPLOT6
C                                                                         007750  SP-CCPLOT7
      ENCODE (100,100,CM(1) )NCASE,ICCPFG                                 007760  SP-CCPLOT8
      CALL COMMNT (CM,10,.FALSE.,.FALSE.)                                 007770  SP-CCPLOT9
C                                                                         007780  SP-CCPLO10
      RETURN                                                              007790  SP-CCPLO11
C                                                                         007800  SP-CCPLO12
C                                                                         007810  SP-CCPLO13
C  * * *  F O R M A T   S T A T E M E N T S  * * *                        007820  SP-CCPLO14
C                                                                         007830  SP-CCPLO15
  100 FORMAT (1H ,20H IN CCPLOT FOR CASE ,I2,14H WITH ICCPFG= ,I1,23H .   007840  SP-CCPLO16
     1 NO PLOT PRODUCED . )                                               007850  SP-CCPLO17
      END                                                                 007860  SP-CCPLO18

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 19 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE COMMNT (A,N1,USEPC,SLCON3)                                  007880   SP-COMMNT2
C                                                                            007890   SP-COMMNT3
      COMMON /COMENT/ CMENT(13,20), COMBUF(13), ICMNT, NTPC                  000190   CM*COMENT2
C                                                                            007910   SP-COMMNT5
      DIMENSION A(N1)                                                        007920   SP-COMMNT6
      LOGICAL USEPC, SLCON3                                                  007930   SP-COMMNT7
C                                                                            007940   SP-COMMNT8
      IF (.NOT.USEPC) GO TO 110                                              007950   SP-COMMNT9
      IF (A(2).NE.1H$) GO TO 140                                             007960   SP-COMMN10
C                                                                            007970   SP-COMMN11
      IF (A(3).EQ.1H$) GO TO 100                                             007980   SP-COMMN12
C                                                                            007990   SP-COMMN13
      ICMNT=ICMNT+1                                                          008000   SP-COMMN14
      ENCODE (120,150,CMENT(1,ICMNT) )(A(I),I=3,80)                          008010   SP-COMMN15
      GO TO 130                                                              008020   SP-COMMN16
C                                                                            008030   SP-COMMN17
  100 IF (NTPC.EQ.0) WRITE (6,160)                                           008040   SP-COMMN18
      WRITE (6,150) (A(I),I=4,80)                                            008050   SP-COMMN19
      NTPC=NTPC+1                                                            008060   SP-COMMN20
      GO TO 140                                                              008070   SP-COMMN21
C                                                                            008080   SP-COMMN22
  110 IF (SLCON3) GO TO 120                                                  008090   SP-COMMN23
      ICMNT=ICMNT+1                                                          008100   SP-COMMN24
      ENCODE (120,170,CMENT(1,ICMNT) )(A(I),I=1,10)                          008110   SP-COMMN25
      GO TO 130                                                              008120   SP-COMMN26
C                                                                            008130   SP-COMMN27
  120 IF (NTPC.EQ.0) WRITE (6,160)                                           008140   SP-COMMN28
      WRITE (6,170) (A(I),I=1,10)                                            008150   SP-COMMN29
      NTPC=NTPC+1                                                            008160   SP-COMMN30
      GO TO 140                                                              008170   SP-COMMN31
C                                                                            008180   SP-COMMN32
  130 IF (ICMNT.LE.20) GO TO 140                                             008190   SP-COMMN33
      WRITE (6,180) (K,(CMENT(J,K),J=1,13),K=1,21)                           008200   SP-COMMN34
      STOP ' IN COMMNT: TOO MANY $$ COMMENTS '                               008210   SP-COMMN35
C                                                                            008220   SP-COMMN36
  140 RETURN                                                                 008230   SP-COMMN37
C                                                                            008240   SP-COMMN38
C                                                                            008250   SP-COMMN39
C  * * *   F O R M A T    S T A T E M E N T S   * * *                        008260   SP-COMMN40
C                                                                            008270   SP-COMMN41
  150 FORMAT (12H ** USER ** ,80A1)                                          008280   SP-COMMN42
  160 FORMAT (1H1,5(8H$$$ $$$ ),4X,30HCOMMENTS FOR THIS SET OF CASES,4X,008290   SP-COMMN43
     15(8H$$$ $$$ ),/)                                                       008300   SP-COMMN44
  170 FORMAT (12H ** COPE ** ,10A10)                                         008310   SP-COMMN45
  180 FORMAT (40H IN COMMNT.  TOO MANY $$ COMMENTS (>20):,/,20(1X,I2,1X,008320   SP-COMMN46
     113A10,/),/,54H THE FOLLOWING COMMENT CAUSED OVERFLOW OF CMENT ARRA008330   SP-COMMN47
     2Y:,/,1X,I2,1X,13A10)                                                   008340   SP-COMMN48
      END                                                                    008350   SP-COMMN49

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 20 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE CREAD (IT2)                                              008370  SP-CREAD 2
C                                                                         008380  SP-CREAD 3
      COMMON /DODF/ DFDOKL(10,3), NDFSP, NDOKIL, DOKILD                   000250  CM+DODF  2
      LOGICAL DOKILD                                                      000260  CM+DODF  3
      COMMON /HIT/ INDEX(6,5,2), PETBL(60,10,7), DLTT(6), PKTBL(10,2),    000410  CM+HIT   2
     1 NPNGTT, RNGTTF(20), TTF(3,20,2), NPNGPS, PNGPST(10), PSTTBL(10,     000420  CM+HIT   3
     2 3)                                                                 000430  CM+HIT   4
      COMMON /RANDOC/ TCRIT, NPLOSP, PNGPLS(10), PPPLOS(10)               000620  CM+RANDOC2
      COMMON /RNGLOS/ NRP, NPP, NPNGCL, CPNGD(11,2), SFGLOS(11,11),       000740  CM+RNGLOS2
     1 RNGCLB(11), VFLTBL(2,2)                                            000750  CM+RNGLOS3
      COMMON /RSPTIM/ RATPTM(3), XHVADF(2), DUM1, RCSPTM(2,2), XHTTIM(2,000790  CM+RSPTIM2
     1 2), TMEAN(2,2), TSIGMA(2,2), DFTTMA(10,2), NDT, TPAPPY(3,2),       000800  CM+RSPTIM3
     2 PPSPTM, TPBAP, PRSDT, PRSVT, ALPHA(2,2), BETA(2,2), DFTTIM,        000810  CM+RSPTIM4
     3 IDSGTP, PUMC, PCPMSG, PPDOW                                        000820  CM+RSPTIM5
      COMMON /WEATHR/ PPCLCG, PPCFLS(2), PPGCFL(11,2), W(6,11,2),         001060  CM+WEATHR2
     1 VIS(11), CLOUD(6), NCC, NVL, PASQL(6), WNDSPD(3,3), HUMID(2),      001070  CM+WEATHR3
     2 WSUM(2), WCFSUM(2), ICC, IVL                                       001080  CM+WEATHR4
C                                                                         001220  DM+M     2
      DIMENSION WDATA(193), ACQDAT(161), RSPDAT(49), DFSDAT(31)           001230  DM+M     3
      DIMENSION RODATA(22), PEDATA(4260), PSTDAT(41), AINVDA(167)         001240  DM+M     4
C                                                                         001250  DM+M     5
      EQUIVALENCE (WDATA(1),PPCLCG)                                       001260  DM+M     6
      EQUIVALENCE (ACQDAT(1),NPP)                                         001270  DM+M     7
      EQUIVALENCE (RSPDAT(1),RATPTM(1))                                   001280  DM+M     8
      EQUIVALENCE (DFSDAT(1),DFDOKL(1))                                   001290  DM+M     9
      EQUIVALENCE (RODATA(1),TCRIT)                                       001300  DM+M    10
      EQUIVALENCE (PEDATA(1),INDEX(1))                                    001310  DM+M    11
      EQUIVALENCE (PSTDAT(1),NPNGPS)                                      001320  DM+M    12
      EQUIVALENCE (AINVDA(1),DLTT(1))                                     001330  DM+M    13
      DIMENSION A(80), B(8,10)                                            008460  SP-CREAD11
C                                                                         008470  SP-CREAD12
      ITYPE=IT2-20                                                        008480  SP-CREAD13
C                                                                         008490  SP-CREAD14
      GO TO (100,170,200,230,260,280,340,370), ITYPE                      008500  SP-CREAD15
C                                                                         008510  SP-CREAD16
  100 DO 110 I=1,193                                                      008520  SP-CREAD17
  110 WDATA(I)=0.0                                                        008530  SP-CREAD18
      READ (5,420) NCC,NVL                                                008540  SP-CREAD19
      READ (5,430) PPCLCG,(PPCFLS(I),I=1,2)                               008550  SP-CREAD20
      DO 120 J=1,2                                                        008560  SP-CREAD21
      READ (5,430) (PPGCFL(I,J),I=1,11)                                   008570  SP-CREAD22
  120 CONTINUE                                                            008580  SP-CREAD23
      DO 130 K=1,2                                                        008590  SP-CREAD24
      DO 130 J=1,11                                                       008600  SP-CREAD25
      READ (5,430) (W(I,J,K),I=1,NCC)                                     008610  SP-CREAD26
  130 CONTINUE                                                            008620  SP-CREAD27
      READ (5,430) (VIS(I),I=1,NVL)                                       008630  SP-CREAD28
      READ (5,430) (CLOUD(I),I=1,NCC)                                     008640  SP-CREAD29
      READ (5,430) (PASQL(I),I=1,6)                                       008650  SP-CREAD30
      DO 140 J=1,3                                                        008660  SP-CREAD31
      READ (5,430) (WNDSPD(I,J),I=1,3)                                    008670  SP-CREAD32
  140 CONTINUE                                                            008680  SP-CREAD33
      READ (5,430) (HUMID(I),I=1,2)                                       008690  SP-CREAD34
      GO TO 410                                                           008700  SP-CREAD35

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 21 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678990 (READ VERTICALLY)

C                                                                                  008710   SP-CREAD36
  150 DO 160 I=1,161                                                               008720   SP-CREAD37
  160 ACODAT(I)=0.0                                                                008730   SP-CREAD38
      READ (5,440) NRP,NPP,NRNGCL                                                  008740   SP-CREAD39
      NRNGCB=NRNGCL+1                                                              008750   SR-CREAD40
      DO 170 J=1,2                                                                 008760   SR-CREAD41
      READ (5,460) (CRNGD(I,J),I=1,NPP)                                            008770   SR-CREAD42
  170 CONTINUE                                                                     008780   SR-CREAD43
      DO 180 J=1,NRNGCB                                                            008790   SR-CREAD44
      READ (5,460) (SEGLOS(I,J),I=1,NPP)                                           008800   SR-CREAD45
  180 CONTINUE                                                                     008810   SR-CREAD46
      READ (5,460) (RNGCLB(I),I=1,NRNGCL)                                          008820   SR-CREAD47
      DO 190 J=1,2                                                                 008830   SP-CREAD48
      READ (5,460) (VELTBL(I,J),I=1,2)                                             008840   SP-CREAD49
  190 CONTINUE                                                                     008850   SP-CREAD50
      GO TO 410                                                                    008860   SR-CREAD51
C                                                                                  008870   SR-CREAD52
  200 DO 210 I=1,49                                                                008880   SR-CREAD53
  210 RSPDAT(I)=0.0                                                                008890   SR-CREAD54
      READ (5,450) NDT                                                             008900   SP-CREAD55
      DO 220 J=1,2                                                                 008910   SR-CREAD56
      READ (5,460) (DETTMA I,J),I=1,NDT)                                           008920   SP-CREAD57
  220 CONTINUE                                                                     008930   SR-CREAD58
      READ (5,460) (BATRTH(I),I=1,3),(XMVADF(I),I=1,2)                             008940   SR-CREAD59
      READ (5,460) ((BCSPTM(I,J),I=1,2),J=1,2),((XMTTIM(I,J),I=1,2),J=1,008950   SP-CREAD60
     1  2)                                                                         008960   SR-CREAD61
      READ (5,460) ((TMEAN(I,J),I=1,2),J=1,2),((TSIGMA(I,J),I=1,2),J=1, 008970   SR-CREAD62
     1  2)                                                                         008980   SP-CREAD63
      READ (5,460) ((TRARRY(I,J),I=1,3),J=1,2)                                     008990   SR-CREAD64
      GO TO 410                                                                    009000   SR-CREAD65
C                                                                                  009010   SP-CREAD66
  230 DO 240 I=1,31                                                                009020   SR-CREAD67
  240 DFSDAT(I)=0.0                                                                009030   SR-CREAD68
      READ (5,450) NDFSP                                                           009040   SP-CREAD69
      DO 250 J=1,3                                                                 009050   SR-CREAD70
      READ (5,460) (DFDDKL(I,J),I=1,NDFSP)                                         009060   SP-CREAD71
  250 CONTINUE                                                                     009070   SP-CREAD72
      GO TO 410                                                                    009080   SP-CREAD73
C                                                                                  009090   SR-CREAD74
  260 DO 270 I=1,22                                                                009100   SP-CREAD75
  270 RODATA(I)=0.0                                                                009110   SP-CREAD76
      READ (5,450) NPLOSP                                                          009120   SR-CREAD77
      READ (5,460) TCPIT                                                           009130   SP-CREAD78
      READ (5,460) (PNGPLS(I),I=1,NPLOSP)                                          009140   SR-CREAD79
      READ (5,460) (PPBLOS(I),I=1,NPLOSP)                                          009150   SP-CREAD80
      GO TO 410                                                                    009160   SR-CREAD81
C                                                                                  009170   SP-CREAD82
  280 DO 290 I=1,4260                                                              009180   SP-CREAD83
  290 PFDATA(I)=0.0                                                                009190   SP-CREAD84
      IUNIT=5                                                                      009200   SP-CREAD85
      READ (5,490) (A(I),I=1,80)                                                   009210   SP-CREAD86
      CALL SEPPEC (A,80,B,8,10)                                                    009220   SP-CREAD87
      IF (B(1,1).NE.8HUSETAPE4) GO TO 300                                          009230   SP-CREAD88
      IUNIT=4                                                                      009240   SP-CREAD89

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678990 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 22 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 9 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)

      GO TO 310                                                          009250   SR-CREAD90
  300 ENCODE (20,490,B(1,1) )(A(I),I=1,20)                               009260   SR-CREAD91
      DECODE (20,470,B(1,1) )(INDEX(1,1,K),K=1,2)                        009270   SR-CREAD92
  310 CONTINUE                                                           009280   SR-CREAD93
      DO 320 I=1,6                                                       009290   SR-CREAD94
      ISTART=1                                                           009300   SR-CREAD95
      IF (I.EQ.1.AND.IUNIT.EQ.5) ISTART=2                                009310   SR-CREAD96
      DO 320 J=ISTART,5                                                  009320   SR-CREAD97
      READ (IUNIT,470) (INDEX(I,J,K),K=1,2)                              009330   SR-CREAD98
  320 CONTINUE                                                           009340   SR-CREAD99
      DO 330 I=1,60                                                      009350   SR-CREA100
      DO 330 J=1,10                                                      009360   SR-CREA101
      READ (IUNIT,480) (PETBL(I,J,K),K=1,7)                              009370   SR-CREA102
  330 CONTINUE                                                           009380   SR-CREA103
      GO TO 410                                                          009390   SR-CREA104
C                                                                        009400   SR-CREA105
  340 DO 350 I=1,41                                                      009410   SR-CREA106
  350 PSTDAT(I)=0.0                                                      009420   SR-CREA107
      READ (5,450) NRNGPS                                                009430   SR-CREA108
      READ (5,460) (RNGPST(I),I=1,NRNGPS)                                009440   SR-CREA109
      DO 360 J=1,3                                                       009450   SR-CREA110
      READ (5,460) (PSTTBL(I,J),I=1,NRNGPS)                              009460   SR-CREA111
  360 CONTINUE                                                           009470   SR-CREA112
      GO TO 410                                                          009480   SR-CREA113
C                                                                        009490   SR-CREA114
  370 DO 380 I=1,167                                                     009500   SR-CREA115
  380 AINVDA(I)=0.0                                                      009510   SR-CREA116
      READ (5,450) NRNGTT                                                009520   SR-CREA117
      DO 390 J=1,NRNGTT                                                  009530   SR-CREA118
      READ (5,460) RNGTTF(J),((TTF(I,J,K),K=1,2),I=1,3)                  009540   SR-CREA119
  390 CONTINUE                                                           009550   SR-CREA120
      READ (5,460) (DLTT(I),I=1,6)                                       009560   SR-CREA121
      DO 400 J=1,2                                                       009570   SR-CREA122
      READ (5,460) (PKTBL(I,J),I=1,7)                                    009580   SR-CREA123
  400 CONTINUE                                                           009590   SR-CREA124
      GO TO 410                                                          009600   SR-CREA125
C                                                                        009610   SR-CREA126
  410 RETURN                                                             009620   SR-CREA127
C                                                                        009630   SR-CREA128
C                                                                        009640   SR-CREA129
C  * * *  F O R M A T   S T A T E M E N T S  * * *                       009650   SR-CREA130
C                                                                        009660   SR-CREA131
  420 FORMAT (2I5)                                                       009670   SR-CREA132
  430 FORMAT (8F10.4)                                                    009680   SR-CREA133
  440 FORMAT (3I5)                                                       009690   SR-CREA134
  450 FORMAT (I5)                                                        009700   SR-CREA135
  460 FORMAT (8F10.4)                                                    009710   SR-CREA136
  470 FORMAT (2I10)                                                      009720   SR-CREA137
  480 FORMAT (7F10.5)                                                    009730   SR-CREA138
  490 FORMAT (80A1)                                                      009740   SR-CREA139
      END                                                                009750   SR-CREA140

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 9 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 23 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE DETCTN                                                 009770  SP-DETCTN2
C                                                                       009780  SP-DETCTN3
      COMMON /FLAG/ ENDREP, IDN, IPREPL, IDIGTL, IPRTY, IPNDOC, IT3FLG, 000300  CM+FLAG  2
     1  ENDFLG                                                          000310  CM+FLAG  3
      LOGICAL ENDFLG, ENDREP                                            000320  CM+FLAG  4
      COMMON /KTEST/ KTEST(20,6)                                        000490  CM+KTEST 2
      COMMON /RANDOM/ IR1, IR2, IR3, IR4, IR5, IR6, IR7, IR8, IR9, IR10,000640  CM+RANDOM2
     1  IR11, IR12, IR13, IR14, IR15, IR16, IR17, IR18, IR19, IR20, IR21000650  CM+RANDOM3
     2  , IR22, IR23, IR24, IR25, IR26, IR27, IR28                      000660  CM+RANDOM4
      COMMON /RPLCTN/ KREP, NREP, KRF, NRF, NVEHKL                      000770  CM+RPLCTN2
      COMMON /RSPTIM/ BATRTM(3), XMVADF(2), DUM1, BCSPTM(2,2), XMTTIM(2,000790  CM+RSPTIM2
     1  2), TMEAN(2,2), TSIGMA(2,2), DETTMA(10,2), NDT, TRAPPY(3,2),    000800  CM+RSPTIM3
     2  PRSPTM, TRBAP, PRSDT, PRSVT, ALPHA(2,2), BETA(2,2), DETTIM,     000810  CM+RSPTIM4
     3  IDSGTP, RUMC, PCRMSG, PRDOW                                     000820  CM+RSPTIM5
      COMMON /TIME/ DURLOS, TIMNOW, TOF, TBR, TIMPA, TIMPA1             001020  CM+TIME  2
C                                                                       009850  SP-DETCT10
      CALL SMPLCD (DETTMA,10,2,2,10,IR8,DETTIM,ICLASS)                  009860  SP-DETCT11
C                                                                       009870  SP-DETCT12
      IF (IPNDOC.EQ.1) GO TO 100                                        009880  SP-DETCT13
C                                                                       009890  SP-DETCT14
      KTEST(4,KRF)=KTEST(4,KRF)+1                                       009900  SP-DETCT15
      IF (DETTIM.LT.DURLOS) GO TO 100                                   009910  SP-DETCT16
C                                                                       009920  SP-DETCT17
      IABRT=4                                                           009930  SP-DETCT18
      CALL ABRTTL (IABRT)                                               009940  SP-DETCT19
C                                                                       009950  SP-DETCT20
  100 RETURN                                                            009960  SP-DETCT21
      END                                                               009970  SP-DETCT22

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 24 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE DEFAULT                                                      009990  SP-DFAULT2
C                                                                             010000  SP-DFAULT3
      COMMON /DVALUE/ D(70), DPPT, DDT, DIDN, DIDIGT, DSMK(2)                 000280  CM+DVALUE2
      COMMON /PUNDAT/ DAT, TIM, NCASE, ICCPFG, IPPFG                          000840  CM+PUNDAT2
C                                                                             010030  SP-DFAULT6
      ICCPFG=0                                                                010040  SP-DFAULT7
      IPPFG=0                                                                 010050  SP-DFAULT8
      D(11)=1.                                                                010060  SP-DFAULT9
      D(12)=1.                                                                010070  SP-DFAUL10
      D(13)=1.                                                                010080  SP-DFAUL11
      D(14)=1.                                                                010090  SP-DFAUL12
      D(16)=1.                                                                010100  SP-DFAUL13
      D(17)=1.                                                                010110  SP-DFAUL14
      D(21)=10HJUN1400021                                                     010120  SP-DFAUL15
      D(22)=10HACQDAT0122                                                     010130  SP-DFAUL16
      D(23)=10HRSPDAT0123                                                     010140  SP-DFAUL17
      D(24)=10HDESDAT0124                                                     010150  SP-DFAUL18
      D(27)=10HPSTDAT0127                                                     010160  SP-DFAUL19
      D(28)=10HAINVDA0128                                                     010170  SP-DFAUL20
      D(31)=0.0                                                               010180  SP-DFAUL21
      D(32)=.40                                                               010190  SP-DFAUL22
      D(41)=.01                                                               010200  SP-DFAUL23
      D(42)=3.                                                                010210  SP-DFAUL24
      D(43)=0.                                                                010220  SP-DFAUL25
      D(44)=1000.                                                             010230  SP-DFAUL26
      D(45)=31.0                                                              010240  SP-DFAUL27
      D(46)=25.                                                               010250  SP-DFAUL28
      D(47)=.37                                                               010260  SP-DFAUL29
      D(48)=.975                                                              010270  SP-DFAUL30
      D(49)=.915                                                              010280  SP-DFAUL31
      D(50)=.96                                                               010290  SP-DFAUL32
      D(51)=20.                                                               010300  SP-DFAUL33
      D(52)=10000.                                                            010310  SP-DFAUL34
      D(53)=6.                                                                010320  SP-DFAUL35
      D(54)=10.                                                               010330  SP-DFAUL36
      D(55)=61.                                                               010340  SP-DFAUL37
      D(56)=8.                                                                010350  SP-DFAUL38
      D(57)=.10                                                               010360  SP-DFAUL39
      D(58)=0.0                                                               010370  SP-DFAUL40
      D(59)=0.0                                                               010380  SP-DFAUL41
CSEC  THE FOLLOWING CARD CONTAINS AN INCORRECT SEEKER SENSITIVITY             010390  SP-DFAUL42
CSEC  DEFAULT VALUE TO AVOID MAKING THE PROGRAM LISTING CLASSIFIED.           010400  SP-DFAUL43
CSEC  WHEN RUNNING IN A SECURE ENVIRONMENT, THE CORRECT VALUE SHOULD BE       010410  SP-DFAUL44
CSEC  USED.  RUNS MADE WITH THE VALUE BELOW ARE USED ONLY FOR THE SAMPLE      010420  SP-DFAUL45
CSEC  CASES.                                                                  010430  SP-DFAUL46
      D(60)=60.                                                               010440  SP-DFAUL47
      D(61)=.975                                                              010450  SP-DFAUL48
      D(62)=.975                                                              010460  SP-DFAUL49
      DPPT=0.0                                                                010470  SP-DFAUL50
      DDT=1.                                                                  010480  SP-DFAUL51
      DIDN=1.                                                                 010490  SP-DFAUL52
      DIDIGT=1.                                                               010500  SP-DFAUL53
      DSMK(1)=484.                                                            010510  SP-DFAUL54
      DSMK(2)=846.                                                            010520  SP-DFAUL55

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 25 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888899 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
C                                                                    010530  SR-DFAUL56
      RETURN                                                         010540  SR-DFAUL57
      END                                                            010550  SR-DFAUL58
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888899 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1  FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 26 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
      SUBROUTINE DFCHK                                                 010570  SR-DFCHK 2
C                                                                      010580  SP-DFCHK 3
      COMMON /DODF/ DFDOKL(10,3), NDFSP, NDOKIL, DOKILD                000250  CM+DODF  2
      LOGICAL DOKILD                                                   000260  CM+DODF  3
      COMMON /KTEST/ KTEST(20,6)                                       000490  CM+KTEST 2
      COMMON /RANDOM/ IP1, IP2, IP3, IP4, IP5, IP6, IP7, IP8, IP9, IP10,000640  CM+RANDOM2
     1  IP11, IR12, IP13, IP14, IP15, IP16, IP17, IP18, IP19, IP20, IR21000650  CM+RANDOM3
     2  , IP22, IR23, IP24, IP25, IP26, IP27, IP28                      000660  CM+RANDOM4
      COMMON /RANGE/ DTRNG, MXDRNG, VISLIM, RNGNOW                     000680  CM+RANGE 2
      REAL MXDRNG                                                      000690  CM+RANGE 3
      COMMON /RPLCTN/ KREP, NREP, KPF, NPF, NVEHKL                     000770  CM+RPLCTN2
      COMMON /TARGET/ TVEL, NVEHCL, DISPVH, ITGTTP, IMUTS, ITGTPS      001000  CM+TARGET2
      COMMON /TIME/ DUPLOS, TIMNOW, TOF, TBP, TIMPA, TIMPA1            001020  CM+TIME  2
      COMMON /WEATHR/ PRCLCG, PRCFLS(2), PRGCFL(11,2), W(6,11,2),      001060  CM+WEATHR2
     1  VIS(11), CLOUD(6), NCC, NVL, PASQL(6), WNDSPD(3,3), HUMID(2),  001070  CM+WEATHR3
     2  WSUM(2), WCFSUM(2), ICC, IVL                                   001080  CM+WEATHR4
C                                                                      010670  SP-DFCHK12
      KTEST(14,KRF)=KTEST(14,KRF)+1                                    010680  SP-DFCHK13
      IF (DOKILD) GO TO 120                                            010690  SP-DFCHK14
C                                                                      010700  SP-DFCHK15
      RNGNOW=AMAX1(0.0,DTRNG-TVEL*TIMPA)                               010710  SR-DFCHK16
      DO 100 I=1,NDFSP                                                 010720  SP-DFCHK17
      IF (DFDOKL(I,1).LE.RNGNOW.AND.RNGNOW.LT.DFDOKL(I+1,1)) GO TO 110 010730  SP-DFCHK18
  100 CONTINUE                                                         010740  SR-DFCHK19
      STOP ' IN DFCHK: ERROR NUMBER 1 '                                010750  SP-DFCHK20
C                                                                      010760  SP-DFCHK21
  110 FRAC=(RNGNOW-DFDOKL(I,1))/(DFDOKL(I+1,1)-DFDOKL(I,1))            010770  SP-DFCHK22
      PDOSUP=DFDOKL(I,2)+FRAC*(DFDOKL(I+1,2)-DFDOKL(I,2))              010780  SR-DFCHK23
      PDOKIL=DFDOKL(I,3)+FRAC*(DFDOKL(I+1,3)-DFDOKL(I,3))              010790  SP-DFCHK24
C                                                                      010800  SP-DFCHK25
      RN15=URAN31(IR15)                                                010810  SR-DFCHK26
      IF (RN15.GT.(PDOSUP+PDOKIL)) GO TO 130                           010820  SP-DFCHK27
      IF (RN15.GT.PDOKIL) GO TO 120                                    010830  SP-DFCHK28
      NDOKIL=NDOKIL+1                                                  010840  SR-DFCHK29
      DOKILD=.TRUE.                                                    010850  SP-DFCHK30
  120 CONTINUE                                                         010860  SP-DFCHK31
C                                                                      010870  SP-DFCHK32
      IABRT=14                                                         010880  SP-DFCHK33
      CALL ABRTTL (IABRT)                                              010890  SP-DFCHK34
C                                                                      010900  SR-DFCHK35
  130 RETURN                                                           010910  SP-DFCHK36
C                                                                      010920  SP-DFCHK37
      END                                                              010930  SP-DFCHK38
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 27 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE DUST                                                        010950   SP-DUST  2
C                                                                            010960   SP-DUST  3
      COMMON /KTEST/ KTEST(20,6)                                             000490   CM+KTEST 2
      COMMON /MISC/ DOARPK, POSTKL, RELIF                                    000540   CM+MISC  2
      COMMON /RANDOM/ IR1, IR2, IR3, IR4, IR5, IR6, IR7, IR8, IR9, IR10,000640   CM+RANDOM2
     1  IR11, IR12, IR13, IR14, IR15, IR16, IR17, IR18, IR19, IR20, IR21000650   CM+RANDOM3
     2  , IR22, IR23, IR24, IR25, IR26, IR27, IR28                           000660   CM+RANDOM4
      COMMON /RPLCTN/ KREP, NREP, KPF, NPF, NVEHKL                           000770   CM+RPLCTN2
C                                                                            011010   SP-DUST  8
      RN14=URAN31(IR14)                                                      011020   SP-DUST  9
C                                                                            011030   SP-DUST 10
      KTEST(6,KPF)=KTEST(6,KPF)+1                                            011040   SP-DUST 11
      IF (RN14.GT.POSTKL) GO TO 100                                          011050   SP-DUST 12
C                                                                            011060   SP-DUST 13
      IABRT=6                                                                011070   SP-DUST 14
      CALL ABPTTL (IABRT)                                                    011080   SP-DUST 15
C                                                                            011090   SP-DUST 16
  100 RETURN                                                                 011100   SP-DUST 17
C                                                                            011110   SP-DUST 18
      END                                                                    011120   SP-DUST 19

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 28 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE ECHO (IUNIT)                                            011140  SP-ECHO  2
C                                                                        011150  SP-ECHO  3
      COMMON /BAIL/ BLDRNG, IBAIL                                        000170  CM+BAIL  2
      COMMON /DODF/ DFDOKL(10,3), NDFSP, NDOKIL, DOKILD                  000250  CM+DODF  2
      LOGICAL DOKILD                                                     000260  CM+DODF  3
      COMMON /FLAG/ ENDPEP, IDN, IPREPL, IDIGIL, IPPTY, IPNDUC, IT3FLG,  000300  CM+FLAG  2
     1  ENDFLG                                                           000310  CM+FLAG  3
      LOGICAL ENDFLG, ENDPEP                                             000320  CM+FLAG  4
      COMMON /HIT/ INDEX(6,5,2), PETBL(60,10,7), DLTT(6), PKTBL(10,2),   000410  CM+HIT   2
     1  NPNGTT, RNGTTF(20), TTF(3,20,2), NPNGPS, PNGPST(10), PSTTBL(10,  000420  CM+HIT   3
     2  3)                                                               000430  CM+HIT   4
      COMMON /LOGFLG/ FIRSTL, SEQNML, SPCL(70), SHPTEC, FTSILL           000510  CM+LOGFLG2
      LOGICAL FIRSTL, SEQNML, SPCL, SHPTEC, FTSILL                       000520  CM+LOGFLG3
      COMMON /MISC/ DOAFPK, PDSTYI, PELIF                                000540  CM+MISC  2
      COMMON /OVER/ IOVER(70)                                            000560  CM+OVER  2
      COMMON /PEDESC/ ANGLET, GTPNG, PEFL, DEFB, TGTHD, SKSFN            000580  CM+PEDESC2
      COMMON /PANDOC/ TCRIT, NPLOSP, PNGPLS(10), PPBLOS(10)              000620  CM+RANDOC2
      COMMON /RANGE/ DTPNG, MXDPNG, VISLIM, PNGNOW                       000680  CM+RANGE 2
      REAL MXDPNG                                                        000690  CM+RANGE 3
      COMMON /RECNAM/ WCODE, ACCODE, PSCODE, DECODE, POCODE, PECODE,     000710  CM+RECNAM2
     1  PSCODE, AICODF                                                   000720  CM+RECNAM3
      COMMON /PNGLOS/ NPP, NPP, NPNGCL, CPNGD(11,2), SEGLOS(11,11),      000740  CM+RNGLOS2
     1  PNGCLB(11), VELTBL(2,2)                                          000750  CM+RNGLOS3
      COMMON /PPLCTN/ KREP, NPEP, KRF, NRF, NVEHKL                       000770  CM+PPLCTN2
      COMMON /PSPTIM/ BATRTM(3), XMVADF(2), DUM1, BCSPTM(2,2), XMTTIM(2, 000790  CM+RSPTIM2
     1  2), TMEAN(2,2), TSIGMA(2,2), DETTMA(10,2), NDT, TPARPY(3,2),     000800  CM+RSPTIM3
     2  PRSPTM, TRBAR, PRSDT, PRSVT, ALPHA(2,2), BETA(2,2), DETTIM,      000810  CM+RSPTIM4
     3  IDSGTP, PUMC, PCPNSG, PPDOW                                      000820  CM+RSPTIM5
      COMMON /PUNDAT/ DAT, TIM, NCASE, ICCPEG, IPPEG                     000840  CM+RUNDAT2
      COMMON /SMOKED/ SMK2(3,3,2), SMK5(3,3,2), FRONT, NSMK2, NSMK5,     000920  CM+SMOKED2
     1  PSMKKL                                                           000930  CM+SMOKED3
      COMMON /TARGET/ TVEL, NVEHCL, DISRVH, ITGTTP, IMUTS, ITGTPS        001000  CM+TARGET2
      COMMON /TIME/ DUELOS, TIMNOW, TOF, TBP, TIMPA, TIMPA)              001020  CM+TIME  2
      COMMON /WEATHR/ PRCLCG, PPCFLS(2), PPGCFL(11,2), W(6,11,2),        001060  CM+WEATHR2
     1  VIS(11), CLOUD(6), NCC, NVL, PASOL(6), WNDSPD(3,3), HUMID(2),    001070  CM+WEATHR3
     2  WSUM(2), WCFSUM(2), ICC, IVL                                     001080  CM+WEATHR4
C                                                                        011350  SP-ECHO 23
      WRITE (IUNIT,200) NCASE                                            011360  SP-ECHO 24
      WRITE (IUNIT,210) NREP,NRF                                         011370  SP-ECHO 25
      WRITE (IUNIT,220) DOAPPK,PSMKKL,PDSTYI,PELIF                       011380  SP-ECHO 26
      WRITE (IUNIT,210) NSMK2,NSMK5                                      011390  SP-ECHO 27
      WRITE (IUNIT,220) MXDPNG                                           011400  SP-ECHO 28
      WRITE (IUNIT,220) TOF,TBP                                          011410  SP-ECHO 29
      WRITE (IUNIT,230) TVEL,NVEHCL,DISRVH,ITGTTP                        011420  SP-ECHO 30
      WRITE (IUNIT,220) PRCLCG,(PPCFLS(I),I=1,2)                         011430  SP-ECHO 31
      WRITE (IUNIT,240) ((PPGCFL(I,J),I=1,11),J=1,2)                     011440  SP-ECHO 32
      DO 100 K=1,2                                                       011450  SP-ECHO 33
      DO 100 J=1,NVL                                                     011460  SP-ECHO 34
      WRITE (IUNIT,220) (W(I,J,K),I=1,NCC)                               011470  SP-ECHO 35
  100 CONTINUE                                                           011480  SP-ECHO 36
      WRITE (IUNIT,210) NCC,NVL                                          011490  SP-ECHO 37
      WRITE (IUNIT,220) (WCFSUM(I),I=1,2),(WSUM(I),I=1,2)                011500  SP-ECHO 38
      WRITE (IUNIT,220) (PASOL(I),I=1,6)                                 011510  SP-ECHO 39
      DO 110 J=1,3                                                       011520  SP-ECHO 40

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 29 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      WRITE (IUNIT,220) (WNDSPD(I,J),I=1,3)                             011530   SP-ECHO 41
  110 CONTINUE                                                          011540   SP-ECHO 42
      WRITE (IUNIT,220) (HUMID(I),I=1,2)                                011550   SP-ECHO 43
      WRITE (IUNIT,210) NRP,NPP,NRNGCL                                  011560   SP-ECHO 44
      WRITE (IUNIT,240) ((CRNGD(I,J),I=1,11),J=1,2)                     011570   SP-ECHO 45
      WRITE (IUNIT,240) ((SFGLOS(I,J),I=1,11),J=1,11)                   011580   SP-ECHO 46
      WRITE (IUNIT,240) (RNGCLB(I),I=1,11)                              011590   SP-ECHO 47
      DO 120 J=1,2                                                      011600   SP-ECHO 48
      WRITE (IUNIT,220) (VELTBL(I,J),I=1,2)                             011610   SP-ECHO 49
  120 CONTINUE                                                          011620   SP-ECHO 50
      WRITE (IUNIT,210) IDN,IPPEPL,IDIGTL,IPRTY,IPNDOC,IT3FLG           011630   SP-ECHO 51
      WRITE (IUNIT 220) BLORNG                                          011640   SR-ECHO 52
      WRITE (IUNIT,250) ((DFDOKL(I,J),I=1,10),J=1,3)                    011650   SP-ECHO 53
      WRITE (IUNIT,250) ANGLET,GTRNG,PEFL,DFFR,TGTHP,SKSEN              011660   SP-ECHO 54
      IF (IOVER(26).NE.2) WRITE (IUNIT,180) RECODE                      011670   SP-ECHO 55
      IF (IOVER(26).EQ.2) WRITE (IUNIT,190)                             011680   SP-ECHO 56
      IF (SHRTEC) GO TO 150                                             011690   SP-ECHO 57
      DO 130 I=1,6                                                      011700   SR-ECHO 58
      WRITE (IUNIT,260) I,((INDEX(I,J,K),K=1,2),J=1,5)                  011710   SR-ECHO 59
  130 CONTINUE                                                          011720   SP-ECHO 60
      DO 140 I=1,60                                                     011730   SP-ECHO 61
      WRITE (IUNIT,270) I,((PFTBL(I,J,K),K=1,7),J=1,10)                 011740   SP-ECHO 62
  140 CONTINUE                                                          011750   SR-ECHO 63
  150 CONTINUE                                                          011760   SP-ECHO 64
      WRITE (IUNIT,210) NRNGTT                                          011770   SP-ECHO 65
      DO 160 J=1,20                                                     011780   SP-ECHO 66
  160 WRITE (IUNIT,220) RNGTTF(J),((TTF(I,J,K),K=1,2),I=1,3)            011790   SP-ECHO 67
      WRITE (IUNIT,220) (DLTT(I),I=1,6)                                 011800   SP-ECHO 68
      WRITE (IUNIT,280) ((PKTBL(I,J),I=1,10),J=1,2)                     011810   SP-ECHO 69
      WRITE (IUNIT,210) NRNGPS                                          011820   SP-ECHO 70
      WRITE (IUNIT,280) (RNGPST(I),I=1,10)                              011830   SR-ECHO 71
      WRITE (IUNIT,280) ((PSTTBL(I,J),I=1,10),J=1,3)                    011840   SP-ECHO 72
      WRITE (IUNIT,220) (BATRTM(I),I=1,3),(XMVADF(I),I=1,2)             011850   SP-ECHO 73
      WRITE (IUNIT,220) ((RCSHTM(I,J),I=1,2),J=1,2),((XMTTIM(I,J),I=1,2)011860   SP-ECHO 74
     1 ,J=1,2)                                                          011870   SP-ECHO 75
      WRITE (IUNIT,220) ((TMEAN(I,J),I=1,2),J=1,2),((TSIGMA(I,J),I=1,2),011880   SR-ECHO 76
     1 J=1,2)                                                           011890   SP-ECHO 77
      WRITE (IUNIT,220) ((ALPHA(I,J),I=1,2),J=1,2),((BETA(I,J),I=1,2),  011900   SR-ECHO 78
     1 J=1,2)                                                           011910   SP-ECHO 79
      WRITE (IUNIT,210) NDT                                             011920   SP-ECHO 80
      WRITE (IUNIT,280) ((DETTMA(I,J),I=1,10),J=1,2)                    011930   SP-ECHO 81
      WRITE (IUNIT,220) ((TPARRY(I,J),I=1,3),J=1,2)                     011940   SP-ECHO 82
      WRITE (IUNIT,220) PPSPTM,TPRAP,PRSDT,PRSVT,PCPMSG,PPDOW,PUMC      011950   SR-ECHO 83
      WRITE (IUNIT,210) IDSGTP                                          011960   SR-ECHO 84
      IF (IPNDOC.EQ.0) GO TO 170                                        011970   SP-ECHO 85
      WRITE (IUNIT,220) TCRIT                                           011980   SP-ECHO 86
      WRITE (IUNIT,210) NPLOSP                                          011990   SP-ECHO 87
      WRITE (IUNIT,280) (RNGPLS(I),I=1,10)                              012000   SP-ECHO 88
      WRITE (IUNIT,280) (PFBLOS(I),I=1,10)                              012010   SP-ECHO 89
  170 CONTINUE                                                          012020   SR-ECHO 90
      WRITE (IUNIT,290) NCASE                                           012030   SP-ECHO 91
C                                                                       012040   SP-ECHO 92
C                                                                       012050   SP-ECHO 93
C  * * *  F O R M A T   S T A T E M E N T S   * * *                     012060   SP-ECHO 94

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1  FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 30 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

C                                                                       012070  SP-ECHO 95
  180 FORMAT (1H ,28H NAME OF PEDATA RECORD USED:,1X,A10)               012080  SP-ECHO 96
  190 FORMAT (1H ,34H TEMPORARY OPTION USED FOR PEDATA )                012090  SP-ECHO 97
  200 FORMAT (1H ,13(4H  - ),21H DATA CHECK FOR CASE ,I3,13(4H  - ))    012100  SP-ECHO 98
  210 FORMAT (1H ,16I5)                                                 012110  SP-ECHO 99
  220 FORMAT (1H ,8F10.4)                                               012120  SP-ECHO100
  230 FORMAT (1H ,F10.4,I10,F10.4,I10)                                  012130  SP-ECHO101
  240 FORMAT (1H ,11F10.4)                                              012140  SP-ECHO102
  250 FORMAT (1H ,10F10.4)                                              012150  SP-ECHO103
  260 FORMAT (1H ,18HINDEX ARRAY FOR I=,I5,/,(7I5))                     012160  SP-ECHO104
  270 FORMAT (1H ,15HPE ARRAY FOR I=,I5,/,(1H ,7F10.4))                 012170  SP-ECHO105
  280 FORMAT (1H ,10F10.4)                                              012180  SP-ECHO106
  290 FORMAT (1H ,8(6H******),27H END OF DATA CHECK FOR CASE,I4,1X,8(6H*012190  SP-ECHO107
     1*****))                                                           012200  SP-ECHO108
      END                                                               012210  SP-ECHO109

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 31 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE FINDIT (B,N1,N2,IT,N3)                                 012230  SP-FINDIT2
C                                                                       012240  SP-FINDIT3
      COMMON /ACHAR/ CHAR(10,120), NREC                                 000150  CM+ACHAR 2
      COMMON /ICODE/ ICODE                                              000450  CM+ICODE 2
      COMMON /INDEX1/ INDX11(2001)                                      000470  CM+INDEX12
      COMMON /PUNDAT/ DAT, TIM, NCASE, ICCREG, IPREG                    000840  CM+PUNDAT2
      COMMON /SYMBOL/ BLANK, DECPNT, ALFBET(27), ANUMBR(11), SEP(5),    000970  CM+SYMBOL2
     1  DOLLAR                                                          000980  CM+SYMBOL3
      EXTERNAL NOREC                                                    012300  SP-FINDIT9
C                                                                       012310  SP-FINDI10
      DIMENSION B(N1,N2), IT(N3)                                        012320  SP-FINDI11
      DIMENSION CHARA(10,30), IPAY(6)                                   012330  SP-FINDI12
C                                                                       012340  SP-FINDI13
      EQUIVALENCE (ICODE,CODE)                                          012350  SP-FINDI14
C                                                                       012360  SP-FINDI15
      LOGICAL NUMRIC                                                    012370  SP-FINDI16
C                                                                       012380  SP-FINDI17
      IPAY(1)=0                                                         012390  SP-FINDI18
      IPAY(5)=LOCF(NOREC)                                               012400  SP-FINDI19
      CALL SYSTEMC (104,IPAY)                                           012410  SP-FINDI20
C                                                                       012420  SP-FINDI21
      DO 100 I=1,N3                                                     012430  SP-FINDI22
  100 IT(I)=1                                                           012440  SP-FINDI23
C                                                                       012450  SP-FINDI24
      IB=1                                                              012460  SP-FINDI25
      ITPY=0                                                            012470  SP-FINDI26
  110 DO 130 I=1,NREC                                                   012480  SP-FINDI27
      DO 120 J=1,8                                                      012490  SP-FINDI28
      IF (B(J,IB).NE.CHAR(J+2,I)) GO TO 130                             012500  SP-FINDI29
  120 CONTINUE                                                          012510  SP-FINDI30
      IT(IB)=CHAR(1,I)                                                  012520  SP-FINDI31
      IMATCH=I                                                          012530  SP-FINDI32
      GO TO 140                                                         012540  SP-FINDI33
  130 CONTINUE                                                          012550  SP-FINDI34
C                                                                       012560  SP-FINDI35
      WRITE (6,240) IB,((B(I1,I2),I1=1,N1),I2=1,N2)                     012570  SP-FINDI36
      STOP ' IN FINDIT: ERROR NUMBER 1 '                                012580  SP-FINDI37
C                                                                       012590  SP-FINDI38
  140 IF (IT(1).LT.6.OR.IT(1).GT.8) GO TO 160                           012600  SP-FINDI39
      IF (ITPY.EQ.0.OR.(20.LT.IT(2).AND.IT(2).LE.30)) GO TO 150         012610  SP-FINDI40
      WRITE (6,230) (B(2,JK),JK=1,N2),IT(2),CHAR(3,IMATCH1)             012620  SP-FINDI41
      STOP ' IN FINDIT: ERROR NUMBER 2 '                                012630  SP-FINDI42
  150 CONTINUE                                                          012640  SP-FINDI43
      IF (ITPY.EQ.1) RETURN                                             012650  SP-FINDI44
      ITPY=1                                                            012660  SP-FINDI45
      IMATCH1=IMATCH                                                    012670  SP-FINDI46
      IB=2                                                              012680  SP-FINDI47
      GO TO 110                                                         012690  SP-FINDI48
C                                                                       012700  SP-FINDI49
  160 IT1=IT(1)                                                         012710  SP-FINDI50
      IF (CHAR(2,IMATCH).EQ.0.0) GO TO 210                              012720  SP-FINDI51
      ICHNL=CHAR(2,IMATCH)                                              012730  SP-FINDI52
      ICHRPL=300                                                        012740  SP-FINDI53
      CODE=CHAR(3,IMATCH)                                               012750  SP-FINDI54

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 32 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      IF (CHAR(1,IMATCH).EQ.CHAR(1,IMATCH+1)) CODE=CHAR(3,IMATCH+1)      012760   SP-FINDI55
      CALL READMS (11,CHARA,ICHPPL,ICODE)                                  012770   SP-FINDI56
      IMXNP=N3                                                             012780   SR-FINDI57
      DO 200 I=2,IMXNP                                                     012790   SP-FINDI58
      IF (B(1,I).EQ.BLANK) GO TO 200                                       012800   SP-FINDI59
      DO 180 K=1,ICHNL                                                     012810   SP-FINDI60
      IF (I.NE.IFIX(CHARA(2,K))) GO TO 180                                 012820   SP-FINDI61
      DO 170 J=1,8                                                         012830   SP-FINDI62
      IF (B(J,I).NE.CHARA(J+2,K)) GO TO 180                                012840   SP-FINDI63
  170 CONTINUE                                                             012850   SP-FINDI64
      IT(I)=CHARA(1,K)                                                     012860   SP-FINDI65
      GO TO 200                                                            012870   SP-FINDI66
  180 CONTINUE                                                             012880   SP-FINDI67
      IB=I                                                                 012890   SP-FINDI68
      IF (NUMRIC(B,N1,N2,IB)) GO TO 190                                    012900   SP-FINDI69
      WRITE (6,240) IB,((B(I1,I2),I1=1,N1),I2=1,N2)                        012910   SP-FINDI70
      STOP ' IN FINDIT: ERROR NUMBER 3 '                                   012920   SP-FINDI71
C                                                                          012930   SP-FINDI72
  190 IT(IB)=0                                                             012940   SP-FINDI73
  200 CONTINUE                                                             012950   SP-FINDI74
C                                                                          012960   SP-FINDI75
      RETURN                                                               012970   SR-FINDI76
  210 IB=2                                                                 012980   SP-FINDI77
      IF (B(1,IB).EQ.BLANK) RETURN                                         012990   SP-FINDI78
      IF (.NOT.NUMRIC(B,N1,N2,IB)) GO TO 220                               013000   SR-FINDI79
      IT(IB)=0                                                             013010   SP-FINDI80
      RETURN                                                               013020   SP-FINDI81
  220 WRITE (6,240) IB,((B(I1,I2),I1=1,N1),I2=1,N2)                        013030   SP-FINDI82
      STOP ' IN FINDIT: ERROR NUMBER 4 '                                   013040   SP-FINDI83
C                                                                          013050   SP-FINDI84
C                                                                          013060   SP-FINDI85
C  * * *   F O R M A T   S T A T E M E N T S   * * *                       013070   SP-FINDI86
C                                                                          013080   SP-FINDI87
  230 FORMAT (1H ,21H IN FINDIT. KEYWORD: ,8A10,/,32H AND CORRESPONDING 013090   SP-FINDI88
     1VALUE OF IT: ,I3,26H ARE NOT ALLOWED WITH THE ,A10,8H OPTION )       013100   SP-FINDI89
  240 FORMAT (1H ,39H IN FINDIT. NO MATCH FOUND FOR LINE IB=,I3,19H OF T013110   SR-FINDI90
     1HIS B ARRAY : ,/,(1X,8A10))                                          013120   SP-FINDI91
      END                                                                  013130   SP-FINDI92

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 33 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE GETRNG                                                 013150  SR-GETRNG2
C                                                                       013160  SR-GETRNG3
      COMMON /RANDOM/ IR1, IR2, IR3, IR4, IR5, IR6, IR7, IR8, IR9, IR10,000640  CM+RANDOM2
     1  IR11, IR12, IR13, IR14, IR15, IR16, IR17, IR18, IR19, IR20, IR21000650  CM+RANDOM3
     2  , IR22, IR23, IR24, IR25, IR26, IR27, IR28                      000660  CM+RANDOM4
      COMMON /RANGE/ DTRNG, MXDRNG, VISLIM, RNGNOW                      000680  CM+RANGE 2
      REAL MXDRNG                                                       000690  CM+RANGE 3
      COMMON /RNGLOS/ NRP, NPP, NRNGCL, CRNGD(11,2), SEGLOS(11,11),     000740  CM+RNGLOS2
     1  RNGCLB(11), VELTBL(2,2)                                         000750  CM+RNGLOS3
      COMMON /RPLCTN/ KREP, NREP, KRF, NRF, NVEHKL                      000770  CM+RPLCTN2
      COMMON /TARGET/ TVEL, NVEHCL, DISBVH, ITGTTP, IMUTS, ITGTPS       001000  CM+TARGET2
      COMMON /TIME/ DUPLOS, TIMNOW, TOF, TBR, TIMPA, TIMPAI             001020  CM+TIME  2
      COMMON /WEATHR/ PRCLCG, PRCFLS(2), PPGCFL(11,2), W(6,11,2),       001060  CM+WEATHR2
     1  VIS(11), CLOUD(6), NCC, NVL, PASQL(6), WNDSPD(3,3), HUMID(2),   001070  CM+WEATHR3
     2  WSUM(2), WCFSUM(2), ICC, IVL                                    001080  CM+WEATHR4
C                                                                       013240  SR-GETRN11
      CALL SMPLCD (CRNGD,11,2,2,NRP,IR6,DTRNG,IRC)                      013250  SR-GETRN12
C                                                                       013260  SR-GETRN13
      DO 100 I=1,NRNGCL                                                 013270  SR-GETRN14
      IF (DTRNG.LE.RNGCLB(I)) GO TO 110                                 013280  SR-GETRN15
  100 CONTINUE                                                          013290  SR-GETRN16
C                                                                       013300  SR-GETRN17
  110 IROW=I+1                                                          013310  SR-GETRN18
C                                                                       013320  SR-GETRN19
      IF (TVEL.GT.0.0) GO TO 120                                        013330  SR-GETRN20
      CALL STRACE                                                       013340  SR-GETRN21
      STOP ' IN GETRNG: TARGET VELOCITY = 0.0 '                         013350  SR-GETRN22
C                                                                       013360  SR-GETRN23
  120 CALL SMPLCD (SEGLOS,11,11,IROW,NPP,IR7,SEGLNG,ICLASS)             013370  SR-GETRN24
      DUPLOS=(SEGLNG+FLOAT(NVEHCL-1)*DISBVH)/TVEL                       013380  SR-GETRN25
C                                                                       013390  SR-GETRN26
      RETURN                                                            013400  SR-GETRN27
C                                                                       013410  SR-GETRN28
      END                                                               013420  SR-GETRN29

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 34 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789O (READ VERTICALLY)

      SUBROUTINE GETTIM                                                   013440  SR-GETTIM2
C                                                                         013450  SR-GETTIM3
      COMMON /FLAG/ ENDREP, IDN, IPRERL, IDIGTL, IPPTY, IPNDMC, IT3FLG,   000300  CM+FLAG  2
     1 ENDFLG                                                             000310  CM+FLAG  3
      LOGICAL ENDFLG, ENDREP                                              000320  CM+FLAG  4
      COMMON /FSRESP/ FSRT(14,2)                                          000360  CM+FSRESP2
      COMMON /KTEST/ KTEST(20,6)                                          000490  CM+KTEST 2
      COMMON /LOGFLG/ FIRSTL, SEQNML, SPCL(70), SHPTEC, FTSILL            000510  CM+LOGFLG2
      LOGICAL FIRSTL, SEQNML, SPCL, SHPTEC, FTSILL                        000520  CM+LOGFLG3
      COMMON /RANDOM/ IR1, IR2, IR3, IR4, IR5, IR6, IR7, IR8, IR9, IR10,  000640  CM+RANDOM2
     1 IR11, IR12, IR13, IR14, IR15, IR16, IR17, IR18, IR19, IR20, IR21   000650  CM+RANDOM3
     2 , IR22, IR23, IR24, IR25, IR26, IR27, IR28                         000660  CM+RANDOM4
      COMMON /RPLCTN/ KREP, NREP, KPF, NPF, NVEHKL                        000770  CM+RPLCTN2
      COMMON /RSPTIM/ EATRTM(3), YMVADF(2), DUM1, BCSPTM(2,2), XMTTIM(2,  000790  CM+RSPTIM2
     1 2), TMEAN(2,2), TSIGMA(2,2), DETTMA(10,2), NDT, TPARPY(3,2),       000800  CM+RSPTIM3
     2 PRSPTM, TRBAR, PPSDT, PPSVT, ALPHA(2,2), BETA(2,2), DETTIM,         000810  CM+RSPTIM4
     3 IDSGTP, RUNC, PCPMSG, PRDOW                                        000820  CM+RSPTIM5
      COMMON /TIME/ DURLDS, TIMNOW, TDE, TBP, TIMPA, TIMRA1               001020  CM+TIME  2
C                                                                         013540  SR-GETTI12
      ITIMDL=3                                                            013550  SR-GETTI13
      IF (FTSILL) ITIMDL=1                                                013560  SR-GETTI14
      IF (PRSPTM.NE.0.0) ITIMDL=2                                         013570  SR-GETTI15
C                                                                         013580  SR-GETTI16
      GO TO (100,110,120), ITIMDL                                         013590  SR-GETTI17
C                                                                         013600  SR-GETTI18
C  FIRST METHOD OF MODELING RESPONSE TIME (A SINGLE DISTRIBUTION)         013610  SR-GETTI19
C                                                                         013620  SR-GETTI20
  100 TIMNOW=DETTIM                                                       013630  SR-GETTI21
C                                                                         013640  SR-GETTI22
      CALL SMPLCD (FSRT,14,2,2,14,IR9,FSPTIM,IC)                          013650  SR-GETTI23
      TIMNOW=TIMNOW+FSPTIM                                                013660  SR-GETTI24
C                                                                         013670  SR-GETTI25
      RN10=URAN31(IR10)                                                   013680  SR-GETTI26
      KTEST(8,KPF)=KTEST(8,KPF)+1                                         013690  SR-GETTI27
      IF (RN10.LE.PPSDT) GO TO 170                                        013700  SR-GETTI28
C                                                                         013710  SR-GETTI29
      IABRT=8                                                             013720  SR-GETTI30
      CALL ABRTTL (IABRT)                                                 013730  SR-GETTI31
      GO TO 180                                                           013740  SR-GETTI32
C                                                                         013750  SR-GETTI33
C  SECOND METHOD OF MODELING RESPONSE TIME (A SINGLE VALUE)               013760  SR-GETTI34
C                                                                         013770  SR-GETTI35
  110 TIMNOW=PRSPTM                                                       013780  SR-GETTI36
C                                                                         013790  SR-GETTI37
      RN10=URAN31(IR10)                                                   013800  SR-GETTI38
      KTEST(8,KPF)=KTEST(8,KPF)+1                                         013810  SR-GETTI39
      IF (RN10.LE.PPSDT) GO TO 170                                        013820  SR-GETTI40
C                                                                         013830  SR-GETTI41
      IABRT=8                                                             013840  SR-GETTI42
      CALL ABRTTL (IABRT)                                                 013850  SR-GETTI43
      GO TO 180                                                           013860  SR-GETTI44
C                                                                         013870  SR-GETTI45
C  THIRD METHOD OF MODELING RESPONSE TIME (ADDING CONTRIBUTIONS OF        013880  SR-GETTI46
C  VARIOUS DELAY COMPONENTS)                                              013890  SR-GETTI47

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789O (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 35 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

C                                                                               013900  SP-GETTI148
  120 CONTINUE                                                                  013910  SP-GETTI149
C                                                                               013920  SP-GETTI150
C  IF IDN = 1 , DAYTIME ; IF IDN = 2 , NIGHT .                                  013930  SP-GETTI151
C  IF IPREPL = 1 , PRE-PLANNED ; IF IPREPL = 2 , TARGET OF OPPORTUNITY .       013940  SP-GETTI152
C  IF IDIGTL = 1 , USE DMD ; IF IDIGTL = 2 , USE VOICE .                        013950  SP-GETTI153
C  IF IPRTY=2, PRIORITY PRE-PLANNED TARGET; OTHERWISE, NOT PRIORITY             013960  SP-GETTI154
C                                                                               013970  SP-GETTI155
      ITDGTL=IDIGTL                                                             013980  SP-GETTI156
      KTEST(8,KPF)=KTEST(8,KPF)+1                                               013990  SP-GETTI157
      IF (ITDGTL.NE.1) GO TO 130                                                014000  SP-GETTI158
C                                                                               014010  SP-GETTI159
      ALF=ALPHA(IDN,IPREPL)                                                     014020  SP-GETTI160
      EDDTIM=BETA(IDN,IPREPL)*GAMMA(ALF,IF9)                                    014030  SP-GETTI161
      TIMNOW=TIMNOW+EDDTIM                                                      014040  SP-GETTI162
C                                                                               014050  SP-GETTI163
      PN10=UPAN31(IR10)                                                         014060  SP-GETTI164
      IF (PN10.GT.PRSDT) GO TO 130                                              014070  SP-GETTI165
      TIMNOW=TIMNOW+XMTTIM(ITDGTL,IPREPL)                                       014080  SP-GETTI166
      GO TO 160                                                                 014090  SP-GETTI167
C                                                                               014100  SP-GETTI168
  130 PN11=UPAN31(IP11)                                                         014110  SP-GETTI169
      IF (PN11.LE.PRSVT) GO TO 140                                              014120  SP-GETTI170
      IABRT=8                                                                   014130  SP-GETTI171
      CALL ABRTTL (IABRT)                                                       014140  SP-GETTI172
      GO TO 180                                                                 014150  SP-GETTI173
C                                                                               014160  SP-GETTI174
  140 IF (IDIGTL.EQ.2) GO TO 150                                                014170  SP-GETTI175
      TIMNOW=TIMNOW+XMTTIM(ITDGTL,IPREPL)+XMVADF(IPREPL)                        014180  SP-GETTI176
      ITDGTL=2                                                                  014190  SP-GETTI177
      GO TO 160                                                                 014200  SP-GETTI178
C                                                                               014210  SP-GETTI179
  150 TIMNOW=TIMNOW+XMTTIM(ITDGTL,IPREPL)                                       014220  SP-GETTI180
  160 TIMNOW=TIMNOW+RCSPTM(ITDGTL,IPREPL)                                       014230  SP-GETTI181
      TIMNOW=TIMNOW+BATPTM(IPREPL+IPRTY)                                        014240  SP-GETTI182
C                                                                               014250  SP-GETTI183
C  CHECK FOR CORRECT MESSAGE                                                    014260  SP-GETTI184
C                                                                               014270  SP-GETTI185
  170 KTEST(9,KPF)=KTEST(9,KPF)+1                                               014280  SP-GETTI186
      PN12=UPAN31(IP12)                                                         014290  SP-GETTI187
      IF (PN12.LT.PCRMSC) GO TO 180                                             014300  SP-GETTI188
C                                                                               014310  SP-GETTI189
      IABRT=9                                                                   014320  SP-GETTI190
      CALL ABRTTL (IABRT)                                                       014330  SP-GETTI191
C                                                                               014340  SP-GETTI192
  180 RETURN                                                                    014350  SP-GETTI193
C                                                                               014360  SP-GETTI194
      END                                                                       014370  SP-GETTI195

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 36 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE GETVIS                                                 014390  SP-GETVIS2
C                                                                       014400  SP-GETVIS3
C  THIS SUBROUTINE SAMPLES WEATHER DATA ARRAY TO OBTAIN CLOUD CEILING   014410  SP-GETVIS4
C  AND VISIBILITY RANGE LIMIT.                                          014420  SP-GETVIS5
C                                                                       014430  SP-GETVIS6
      COMMON /RANDOM/ IP1, IP2, IP3, IP4, IP5, IP6, IP7, IP8, IP9, IP10,000640  CM*RANDOM2
     1  IP11, IP12, IP13, IP14, IP15, IP16, IP17, IP18, IP19, IP20, IP21000650  CM*RANDOM3
     2  , IP22, IP23, IP24, IP25, IP26, IP27, IP28                      000660  CM*RANDOM4
      COMMON /RANGE/ DTRNG, MXDRNG, VISLIM, RNGNOW                      000680  CM*RANGE 2
      REAL MXDRNG                                                       000690  CM*RANGE 3
      COMMON /RPLCTN/ KEEP, NREP, KRF, NRF, NVEHKL                      000770  CM*RPLCTN2
      COMMON /WEATHR/ PRCLCG, PRCFLS(2), PRGCFL(11,2), W(6,11,2),       001060  CM*WEATHR2
     1  VIS(11), CLOUD(6), NCC, NVL, PASQL(6), WNDSPD(3,3), HUMID(2),   001070  CM*WEATHR3
     2  WSUM(2), WCFSUM(2), ICC, IVL                                    001080  CM*WEATHR4
C                                                                       014480  SP-GETVI11
C  IF THERE IS A CLOUD CEILING, ICLNGF=1; IF NOT,ICLNGF=2 .             014490  SP-GETVI12
C                                                                       014500  SP-GETVI13
      RN2=URAN31(IP2)                                                   014510  SP-GETVI14
      ICLNGF=2                                                          014520  SP-GETVI15
      IF (RN2.LT.PRCLCG) ICLNGF=1                                       014530  SP-GETVI16
C                                                                       014540  SP-GETVI17
C  IF THERE IS A CLOUD FREE LOS, ICFLOS=1;IF NOT, ICFLOS=2 .            014550  SP-GETVI18
C                                                                       014560  SP-GETVI19
      RN3=URAN31(IP3)                                                   014570  SP-GETVI20
      ICFLOS=2                                                          014580  SP-GETVI21
      IF (RN3.LT.PRCFLS(ICLNGF)) ICFLOS=1                               014590  SP-GETVI22
      IF (ICFLOS.EQ.2) GO TO 120                                        014600  SP-GETVI23
C                                                                       014610  SP-GETVI24
C  CLOUD FREE LINE-OF-SIGHT                                             014620  SP-GETVI25
C  USE PRGCFL ARRAY PROBABILITIES IN SAMPLING TO OBTAIN IVL             014630  SP-GETVI26
C  SET ICC TO MAXIMUM CLOUD CEILING ALTITUDE INDEX (ICC=6)              014640  SP-GETVI27
C                                                                       014650  SP-GETVI28
      ICC=6                                                             014660  SP-GETVI29
      CLDCLG=CLOUD(ICC)                                                 014670  SP-GETVI30
C                                                                       014680  SP-GETVI31
      PR=0.0                                                            014690  SP-GETVI32
      RN4=URAN31(IP4)*WCFSUM(ICLNGF)                                    014700  SP-GETVI33
      DO 100 IJ=1,NVL                                                   014710  SP-GETVI34
      PR1=PR                                                            014720  SP-GETVI35
      PR=PR+PRGCFL(IJ,ICLNGF)                                           014730  SP-GETVI36
      IF (PR1.LE.RN4.AND.RN4.LT.PR) GO TO 110                           014740  SP-GETVI37
  100 CONTINUE                                                          014750  SP-GETVI38
C                                                                       014760  SP-GETVI39
      CALL STRACE                                                       014770  SP-GETVI40
      STOP ' IN GETVIS: RANDOM NUMBER TROUBLE 1 '                       014780  SP-GETVI41
C                                                                       014790  SP-GETVI42
  110 VISLIM=VIS(IJ)                                                    014800  SP-GETVI43
      IVL=IJ                                                            014810  SP-GETVI44
      GO TO 160                                                         014820  SP-GETVI45
C                                                                       014830  SP-GETVI46
C  NO CLOUD FREE LINE-OF-SIGHT                                          014840  SP-GETVI47
C  USE W ARRAY PROBABILITIES IN SAMPLING TO OBTAIN ICC AND IVL          014850  SP-GETVI48
C                                                                       014860  SP-GETVI49
  120 PR=0.0                                                            014870  SP-GETVI50

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1  FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 37 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789O (READ VERTICALLY)

      RN5=URAN31(IR5)*WSUM(ICLNGE)                                       014880  SP-GETVI51
      DO 140 I=1,NCC                                                     014890  SP-GETVI52
      DO 130 J=1,NVL                                                     014900  SP-GETVI53
      PP1=PP                                                             014910  SP-GETVI54
      PP=PP+W(I,J,ICLNGE)                                                014920  SP-GETVI55
      IF (PP1.LE.RN5.AND.RN5.LT.PP) GO TO 150                            014930  SP-GETVI56
  130 CONTINUE                                                           014940  SP-GETVI57
  140 CONTINUE                                                           014950  SP-GETVI58
C                                                                        014960  SP-GETVI59
      CALL STRACE                                                        014970  SP-GETVI60
      STOP ' IN GETVIS: RANDOM NUMBER TROUBLE 2 '                        014980  SP-GETVI61
C                                                                        014990  SP-GETVI62
  150 VISLIM=VIS(J)                                                      015000  SP-GETVI63
      CLDCLG=CLOUD(I)                                                    015010  SP-GETVI64
      IVL=J                                                              015020  SP-GETVI65
      ICC=I                                                              015030  SP-GETVI66
C                                                                        015040  SP-GETVI67
  160 ICC=7-ICC                                                          015050  SP-GETVI68
C                                                                        015060  SP-GETVI69
      RETURN                                                             015070  SP-GETVI70
C                                                                        015080  SP-GETVI71
      END                                                                015090  SP-GETVI72

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789O (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 38 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 9 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)

      SUBROUTINE HEADER                                                 015110  SP-HEADER2
C                                                                       015120  SP-HEADER3
      COMMON /BAIL/ BLORNG, IBAIL                                       000170  CM+BAIL  2
      COMMON /COMENT/ CMENT(13,20), COMBUF(13), ICMNT, NTPC             000190  CM+COMENT2
      COMMON /DISPLY/ DISPT(24), DISPN(12), PFCLBL(8), TMPLBL(8)        000230  CM+DISPLY2
      COMMON /FLAG/ ENDPER, IDN, IPFFPL, IDIGTL, IPPTY, IPNDOC, IT3FLG, 000300  CM+FLAG  2
     1  ENDFLG                                                          000310  CM+FLAG  3
      LOGICAL ENDFLG, ENDPER                                            000320  CM+FLAG  4
      COMMON /HEADNG/ ACQLBL(2,4), DAYLBL(2), DFSLBL(2), DGTLBL(2,2),   000380  CM+HEADNG2
     1  DULBL(3,2), ROLBL(4,2), PSPLBL(3,3)                             000390  CM+HEADNG3
      COMMON /LOGFLG/ FIRSTL, SEQNML, SPCL(70), SHFTEC, FTSILL          000510  CM+LOGFLG2
      LOGICAL FIRSTL, SEQNML, SPCL, SHPTEC, FTSILL                      000520  CM+LOGFLG3
      COMMON /MISC/ DOAPPK, PDSTKL, PFLIF                               000540  CM+MISC  2
      COMMON /OVER/ IOVER(70)                                           000560  CM+OVER  2
      COMMON /PEDESC/ ANGLFT, GTPNG, PEFL, DEFB, TGTHD, SKSEN           000580  CM+PEDESC2
      COMMON /RANGE/ DTFNG, MXDRNG, VISLIM, RNGNOW                      000680  CM+RANGE 2
      REAL MXDRNG                                                       000690  CM+RANGE 3
      COMMON /RECNAM/ WCODE, ACCODE, RSCODE, DFCODE, POCODE, PFCODE,    000710  CM+RECNAM2
     1  PSCODE, AICODE                                                  000720  CM+RECNAM3
      COMMON /RPLCTN/ KPEP, NREP, KRF, NRF, NVEHKL                      000770  CM+RPLCTN2
      COMMON /RSPTIM/ BATRTM(3), XMVADF(2), DUM1, RCSPTM(2,2), XMTTIM(2,000790  CM+RSPTIM2
     1  2), THEAN(2,2), TSIGMA(2,2), DETTMA(10,2), HDT, TRAPPY(3,2),    000800  CM+RSPTIM3
     2  PPSPTM, TRBAP, PPSDT, PPSVT, ALPHA(2,2), BETA(2,2), DETTIM,     000810  CM+RSPTIM4
     3  IDSGTP, RUMC, PCPMSG, PRDOW                                     000820  CM+RSPTIM5
      COMMON /RUNDAT/ DAT, TIM, NCASE, ICCPFG, IPPFG                    000840  CM+RUNDAT2
      COMMON /SMOKED/ SMK2(3,3,2), SMK5(3,3,2), FRONT, NSMK2, NSMK5,    000920  CM+SMOKED2
     1  PSMKKL                                                          000930  CM+SMOKED3
      COMMON /SYMBOL/ BLANK, DECPNT, ALFBET(27), ANUMBR(11), SEP(5),    000970  CM+SYMBOL2
     1  DOLLAR                                                          000980  CM+SYMBOL3
      COMMON /TARGET/ TVEL, NVEHCL, DISRVH, ITGTTP, IMUTS, ITGTPS       001000  CM+TARGET2
      COMMON /TIME/ DURLOS, TIMNOW, TOF, TBP, TIMPA, TIMPA1             001020  CM+TIME  2
      COMMON /UVALUE/ U(70), UPPT, UDT, UIDN, UIDIGT, USMK(2)           001040  CM+UVALUE2
C                                                                       015320  SP-HEADE23
      DIMENSION AOUT(4,40), BOUT(4,40), COUT(4,40)                      015330  SP-HEADE24
      DIMENSION CM(10), RECODE(8)                                       015340  SP-HEADE25
      DIMENSION UVALUE(70)                                              015350  SP-HEADE26
C                                                                       015360  SP-HEADE27
      EQUIVALENCE (U(1),UVALUE(1))                                      015370  SP-HEADE28
      EQUIVALENCE (RECODE(1),WCODE)                                     015380  SP-HEADE29
C                                                                       015390  SP-HEADE30
      DO 100 I=1,4                                                      015400  SP-HEADE31
      DO 100 J=1,40                                                     015410  SP-HEADE32
      AOUT(I,J)=BLANK                                                   015420  SP-HEADE33
      BOUT(I,J)=BLANK                                                   015430  SP-HEADE34
      COUT(I,J)=BLANK                                                   015440  SP-HEADE35
  100 CONTINUE                                                          015450  SP-HEADE36
C                                                                       015460  SP-HEADE37
      WRITE (6,400) NCASE                                               015470  SP-HEADE38
      WRITE (6,410) NREP                                                015480  SP-HEADE39
C                                                                       015490  SP-HEADE40
      NXTLIN=0                                                          015500  SP-HEADE41
      ENCODE (40,420,AOUT(1,1+NXTLIN) )                                 015510  SP-HEADE42
      ENCODE (40,430,AOUT(1,2+NXTLIN) )                                 015520  SP-HEADE43
      ENCODE (40,440,AOUT(1,3+NXTLIN) )                                 015530  SP-HEADE44

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 9 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 39 Of 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      ENCODE (40,430,AOUT(1,4+NXTLIN) )                                 015540  SP-HEADE45
      ENCODE (40,450,AOUT(1,5+NXTLIN) )ITGTTP                           015550  SP-HEADE46
      ENCODE (40,460,AOUT(1,6+NXTLIN) )NVEHCL                           015560  SP-HEADE47
      ENCODE (40,470,AOUT(1,7+NXTLIN) )NISBVH                           015570  SP-HEADE48
      ENCODE (40,480,AOUT(1,8+NXTLIN) )TVEL                             015580  SP-HEADE49
      NXTLIN=1                                                          015590  SP-HEADE50
      ENCODE (40,290,AOUT(1,NXTLIN+8) )TGTHD                            015600  SP-HEADE51
      NXTLIN=NXTLIN+1                                                   015610  SP-HEADE52
      ENCODE (40,300,AOUT(1,NXTLIN+8) )PFEL                             015620  SP-HEADE53
      ENCODE (40,430,AOUT(1,9+NXTLIN) )                                 015630  SP-HEADE54
      ENCODE (40,420,AOUT(1,10+NXTLIN) )                                015640  SP-HEADE55
C                                                                       015650  SP-HEADE56
      ENCODE (40,420,AOUT(1,12+NXTLIN) )                                015660  SP-HEADE57
      ENCODE (40,430,AOUT(1,13+NXTLIN) )                                015670  SP-HEADE58
      ENCODE (40,490,AOUT(1,14+NXTLIN) )                                015680  SP-HEADE59
      ENCODE (40,430,AOUT(1,15+NXTLIN) )                                015690  SP-HEADE60
      ENCODE (40,500,AOUT(1,16+NXTLIN) )NFF                             015700  SP-HEADE61
      ENCODE (40,510,AOUT(1,17+NXTLIN) )PFLIF                           015710  SP-HEADE62
      ENCODE (40,520,AOUT(1,18+NXTLIN) )TPP                             015720  SP-HEADE63
      ENCODE (40,530,AOUT(1,19+NXTLIN) )TOF                             015730  SP-HEADE64
      NXTLIN=NXTLIN+1                                                   015740  SP-HEADE65
      ENCODE (40,310,AOUT(1,NXTLIN+19) )GTPNG                           015750  SP-HEADE66
      NXTLIN=NXTLIN+1                                                   015760  SP-HEADE67
      ENCODE (40,320,AOUT(1,NXTLIN+19) )ANGLET                          015770  SP-HEADE68
      NXTLIN=NXTLIN+1                                                   015780  SP-HEADE69
      ENCODE (40,330,AOUT(1,NXTLIN+19) )DFFB                            015790  SP-HEADE70
      ENCODE (40,430,AOUT(1,20+NXTLIN) )                                015800  SP-HEADE71
      ENCODE (40,420,AOUT(1,21+NXTLIN) )                                015810  SP-HEADE72
C                                                                       015820  SP-HEADE73
      ENCODE (40,420,AOUT(1,23+NXTLIN) )                                015830  SP-HEADE74
      ENCODE (40,430,AOUT(1,24+NXTLIN) )                                015840  SP-HEADE75
      ENCODE (40,540,AOUT(1,25+NXTLIN) )                                015850  SP-HEADE76
      ENCODE (40,430,AOUT(1,26+NXTLIN) )                                015860  SR-HEADE77
      IF (IT3FLG.EQ.0) GO TO 110                                        015870  SR-HEADE78
      ENCODE (40,550,AOUT(1,27+NXTLIN) )                                015880  SR-HEADE79
      ENCODE (40,560,AOUT(1,28+NXTLIN) )PHMC                            015890  SP-HEADE80
      NXTLIN=NXTLIN+29                                                  015900  SP-HEADE81
      GO TO 120                                                         015910  SP-HEADE82
  110 ENCODE (40,570,AOUT(1,27+NXTLIN) )                                015920  SP-HEADE83
      NXTLIN=NXTLIN+28                                                  015930  SR-HEADE84
  120 CONTINUE                                                          015940  SP-HEADE85
      ENCODE (40,340,AOUT(1,NXTLIN) )SKSEN                              015950  SP-HEADE86
      NXTLIN=NXTLIN+1                                                   015960  SR-HEADE87
      ENCODE (40,430,AOUT(1,NXTLIN) )                                   015970  SR-HEADE88
      NXTLIN=NXTLIN+1                                                   015980  SP-HEADE89
      ENCODE (40,420,AOUT(1,NXTLIN) )                                   015990  SP-HEADE90
C                                                                       016000  SP-HEADE91
      NXTLIN=0                                                          016010  SP-HEADE92
      ENCODE (40,420,BOUT(1,1) )                                        016020  SP-HEADE93
      ENCODE (40,430,BOUT(1,2) )                                        016030  SP-HEADE94
      ENCODE (40,580,BOUT(1,3) )                                        016040  SP-HEADE95
      ENCODE (40,430,BOUT(1,4) )                                        016050  SP-HEADE96
      NXTLIN=5                                                          016060  SR-HEADE97
      IF (PREPTM.EQ.0.0) GO TO 130                                      016070  SP-HEADE98

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 40 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      ENCODE (40,590,BOUT(1,5) )PRSPTM                                        016080  SP-HEAD 99
      NXTLIN=6                                                                016090  SP-HEAD100
  130 ENCODE (40,600,BOUT(1,NXTLIN) )(PSPLBL(I,IPPTPL+IPPTY),I=1,3)           016100  SP-HEAD101
      ENCODE (40,610,BOUT(1,NXTLIN+1) )(DGTLBL(I,IDIGTL),I=1,2)               016110  SP-HEAD102
      ENCODE (40,620,BOUT(1,NXTLIN+2) )(DAYLBL(IDM))                          016120  SP-HEAD103
      NXTLIN=NXTLIN+3                                                         016130  SP-HEAD104
      TRBAP1=TRBAP                                                            016140  SP-HEAD105
      IF (PRSPTM.GT.0.0) TRBAP1=PRSPTM+TOF                                    016150  SP-HEAD106
      ENCODE (40,630,BOUT(1,NXTLIN) )TRBAP1                                   016160  SP-HEAD107
      NXTLIN=NXTLIN+1                                                         016170  SP-HEAD108
      IF (IDIGTL.EQ.2) GO TO 140                                              016180  SP-HEAD109
      ENCODE (40,640,BOUT(1,NXTLIN) )PPSPT                                    016190  SP-HEAD110
      NXTLIN=NXTLIN+1                                                         016200  SP-HEAD111
      IF (FTSILL) GO TO 150                                                   016210  SP-HEAD112
      IF (PPSPTM.GT.0.0) GO TO 150                                            016220  SP-HEAD113
  140 ENCODE (40,650,BOUT(1,NXTLIN) )PPSVT                                    016230  SP-HEAD114
      NXTLIN=NXTLIN+1                                                         016240  SP-HEAD115
  150 ENCODE (40,260,BOUT(1,NXTLIN) )PCPMSG                                   016250  SP-HEAD116
      NXTLIN=NXTLIN+1                                                         016260  SP-HEAD117
      ENCODE (40,270,BOUT(1,NXTLIN) )PPDOW                                    016270  SP-HEAD118
      NXTLIN=NXTLIN+1                                                         016280  SP-HEAD119
      ENCODE (40,430,BOUT(1,NXTLIN) )                                         016290  SP-HEAD120
      NXTLIN=NXTLIN+1                                                         016300  SP-HEAD121
      ENCODE (40,420,BOUT(1,NXTLIN) )                                         016310  SP-HEAD122
C                                                                             016320  SP-HEAD123
      NXTLIN=NXTLIN+2                                                         016330  SP-HEAD124
      ENCODE (40,420,BOUT(1,NXTLIN) )                                         016340  SP-HEAD125
      NXTLIN=NXTLIN+1                                                         016350  SP-HEAD126
      ENCODE (40,430,BOUT(1,NXTLIN) )                                         016360  SP-HEAD127
      ENCODE (40,660,BOUT(1,NXTLIN+1) )                                       016370  SP-HEAD128
      ENCODE (40,430,BOUT(1,NXTLIN+2) )                                       016380  SP-HEAD129
      ENCODE (40,670,BOUT(1,NXTLIN+3) )IDSGTP                                 016390  SP-HEAD130
      ENCODE (40,680,BOUT(1,NXTLIN+4) )MXDPNG                                 016400  SP-HEAD131
      ENCODE (40,690,BOUT(1,NXTLIN+5) )ELDPNG                                 016410  SP-HEAD132
      IDOLOC=UVALUE(13)                                                       016420  SP-HEAD133
      ENCODE (40,700,BOUT(1,NXTLIN+6) )(DOLBL(I,IDOLOC),I=1,3)                016430  SP-HEAD134
      ENCODE (40,430,BOUT(1,NXTLIN+7) )                                       016440  SP-HEAD135
      ENCODE (40,420,BOUT(1,NXTLIN+8) )                                       016450  SP-HEAD136
C                                                                             016460  SP-HEAD137
      ENCODE (40,420,BOUT(1,NXTLIN+10) )                                      016470  SP-HEAD138
      ENCODE (40,430,BOUT(1,NXTLIN+11) )                                      016480  SP-HEAD139
      ENCODE (40,710,BOUT(1,NXTLIN+12) )                                      016490  SP-HEAD140
      ENCODE (40,430,BOUT(1,NXTLIN+13) )                                      016500  SP-HEAD141
      ENCODE (40,720,BOUT(1,NXTLIN+14) )DDAPPY                                016510  SP-HEAD142
      DECODE (10,730,UVALUE(24) )IDEL                                         016520  SP-HEAD143
      ENCODE (40,740,BOUT(1,NXTLIN+15) )DESLPL(IDEL)                          016530  SP-HEAD144
      ENCODE (40,430,BOUT(1,NXTLIN+16) )                                      016540  SP-HEAD145
      ENCODE (40,420,BOUT(1,NXTLIN+17) )                                      016550  SP-HEAD146
C                                                                             016560  SP-HEAD147
      NXTLIN=NXTLIN+1                                                         016570  SP-HEAD148
      ENCODE (40,420,COUT(1,1) )                                              016580  SP-HEAD149
      ENCODE (40,430,COUT(1,2) )                                              016590  SP-HEAD150
      ENCODE (40,750,COUT(1,3) )                                              016600  SP-HEAD151
      ENCODE (40,430,COUT(1,4) )                                              016610  SP-HEAD152

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 41 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      IF (IOVER(21).NE.0) GO TO 160                                       016620   SP-HEAD153
      DECODE (10,760,UVALUE(21) )XWTHP,YWTHP                              016630   SP-HEAD154
      ENCODE (40,770,COUT(1,5) )XWTHP                                     016640   SP-HEAD155
      ENCODE (40,780,COUT(1,6) )YWTHP                                     016650   SP-HEAD156
      NXTLIN=7                                                            016660   SP-HEAD157
      GO TO 170                                                           016670   SP-HEAD158
  160 IF (IOVER(21).EQ.1) ENCODE (40,790,COUT(1,5) )UVALUE(21)            016680   SP-HEAD159
      IF (IOVER(21).EQ.2) ENCODE (40,800,COUT(1,5) )                      016690   SP-HEAD160
      NXTLIN=6                                                            016700   SP-HEAD161
  170 ENCODE (40,430,COUT(1,NXTLIN) )                                     016710   SP-HEAD162
      ENCODE (40,420,COUT(1,NXTLIN+1) )                                   016720   SP-HEAD163
C                                                                         016730   SP-HEAD164
      ENCODE (40,420,COUT(1,NXTLIN+3) )                                   016740   SP-HEAD165
      ENCODE (40,430,COUT(1,NXTLIN+4) )                                   016750   SP-HEAD166
      ENCODE (40,810,COUT(1,NXTLIN+5) )                                   016760   SP-HEAD167
      ENCODE (40,430,COUT(1,NXTLIN+6) )                                   016770   SP-HEAD168
      IF (IOVER(22).NE.0) GO TO 180                                       016780   SP-HEAD169
      DECODE (10,820,UVALUE(22) )ITRPN                                    016790   SP-HEAD170
      ENCODE (40,830,COUT(1,NXTLIN+7) )(ACQLBL(I,ITRPN),I=1,2)            016800   SP-HEAD171
      GO TO 190                                                           016810   SP-HEAD172
  180 IF (IOVER(22).EQ.1) ENCODE (40,840,COUT(1,NXTLIN+7) )UVALUE(22)     016820   SP-HEAD173
      IF (IOVER(22).EQ.2) ENCODE (40,850,COUT(1,NXTLIN+7) )               016830   SP-HEAD174
  190 ENCODE (40,860,COUT(1,NXTLIN+8) )(POLBL(I,IPNDGC+1),I=1,4)          016840   SP-HEAD175
      NXTLIN=NXTLIN-1                                                     016850   SP-HEAD176
      ENCODE (40,430,COUT(1,NXTLIN+10) )                                  016860   SP-HEAD177
      ENCODE (40,420,COUT(1,NXTLIN+11) )                                  016870   SP-HEAD178
C                                                                         016880   SP-HEAD179
      ENCODE (40,420,COUT(1,NXTLIN+13) )                                  016890   SP-HEAD180
      ENCODE (40,430,COUT(1,NXTLIN+14) )                                  016900   SP-HEAD181
      ENCODE (40,870,COUT(1,NXTLIN+15) )                                  016910   SP-HEAD182
      ENCODE (40,430,COUT(1,NXTLIN+16) )                                  016920   SP-HEAD183
      ENCODE (40,350,COUT(1,NXTLIN+17) )PSMKK1                            016930   SP-HEAD184
      IF (SPCL(32)) GO TO 200                                             016940   SP-HEAD185
      NXTLIN=NXTLIN+1                                                     016950   SP-HEAD186
      ENCODE (40,880,COUT(1,NXTLIN+17) )HSMK2,HSMK5                       016960   SP-HEAD187
  200 CONTINUE                                                            016970   SP-HEAD188
      ENCODE (40,890,COUT(1,NXTLIN+18) )PPSTKL                            016980   SP-HEAD189
      ENCODE (40,430,COUT(1,NXTLIN+19) )                                  016990   SP-HEAD190
      ENCODE (40,420,COUT(1,NXTLIN+20) )                                  017000   SP-HEAD191
C                                                                         017010   SP-HEAD192
      ICMFLG=0                                                            017020   SP-HEAD193
      IF (IDN.EQ.1.AND.YWTHP.EQ.4H2200) ICMFLG=1                          017030   SP-HEAD194
      IF (IDN.EQ.2.AND.YWTHP.NE.4H2200) ICMFLG=1                          017040   SP-HEAD195
      IF (ICMFLG.EQ.0) GO TO 210                                          017050   SP-HEAD196
      ENCODE (100,280,CM(1) )DAYLBL(IDN),YWTHP                            017060   SP-HEAD197
      CALL COMMNT (CM,10,.FALSE.,.FALSE.)                                 017070   SP-HEAD198
  210 CONTINUE                                                            017080   SP-HEAD199
C                                                                         017090   SP-HEAD200
      DO 230 I=1,8                                                        017100   SP-HEAD201
      IF (IOVER(20+I).EQ.0) GO TO 230                                     017110   SP-HEAD202
      IF (IOVER(20+I).EQ.1) GO TO 220                                     017120   SP-HEAD203
      IF (IOVER(20+I).NE.2) STOP ' IN HEADER: ERROR NUMBER 1 '            017130   SP-HEAD204
      ENCODE (100,360,CM(1) )THRLBL(I)                                    017140   SP-HEAD205
      CALL COMMNT (CM,10,.FALSE.,.FALSE.)                                 017150   SP-HEAD206

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 42 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
      GO TO 230                                                         017160   SP-HEAD207
  220 ENCODE (100,370,CM(1) )RECODE(I),TMPLBL(I)                        017170   SP-HEAD208
      CALL COMMNT (CM,10,.FALSE.,.FALSE.)                               017180   SP-HEAD209
      IF (I.NE.6) GO TO 230                                             017190   SP-HEAD210
      DECODE (10,380,RECODE(6) )RECHK                                   017200   SP-HEAD211
      IF (RECHK.NE.4HREOO) GO TO 230                                    017210   SP-HEAD212
      CALL REIONT (T1,T2,T3,T4,T5,T6,T7,T8,T9,RECODE)                   017220   SP-HEAD213
      ENCODE (100,390,CM(1) )T1,T2,T3,T4,T5,T6,T7,T8,T9                 017230   SP-HEAD214
      CALL COMMNT (CM,10,.FALSE.,.FALSE.)                               017240   SP-HEAD215
  230 CONTINUE                                                          017250   SP-HEAD216
C                                                                       017260   SP-HEAD217
      DO 240 J=1,36                                                     017270   SP-HEAD218
      WRITE (6,900) (AOUT(I,J),I=1,4),(BOUT(I,J),I=1,4),(COUT(I,J),I=1, 017280   SP-HEAD219
     1 4)                                                               017290   SP-HEAD220
  240 CONTINUE                                                          017300   SP-HEAD221
      WRITE (6,910)                                                     017310   SP-HEAD222
      IF (ICMNT.NE.0) WRITE (6,920)                                     017320   SP-HEAD223
      DO 250 J=1,ICMNT                                                  017330   SP-HEAD224
      WRITE (6,930) (CMENT(I,J),I=1,13)                                 017340   SP-HEAD225
  250 CONTINUE                                                          017350   SP-HEAD226
C                                                                       017360   SP-HEAD227
      RETURN                                                            017370   SP-HEAD228
C                                                                       017380   SP-HEAD229
C                                                                       017390   SP-HEAD230
C  * * *   F O R M A T   S T A T E M E N T S   * * *                    017400   SP-HEAD231
C                                                                       017410   SP-HEAD232
  260 FORMAT (1H*,23H PROB CORRECT MESSAGE =,F6.3,9X,1H*)               017420   SP-HEAD233
  270 FORMAT (1H*,25H PROB DO WARNED TO LASE =,F6.3,7X,1H*)             017430   SP-HEAD234
  280 FORMAT (1H ,10HCHOICE OF ,A10,53H FOR RESPONSE TIME DATA IS INCONS017440   SP-HEAD235
     1ISTENT WITH TIME OF ,A4,18H FOR WEATHER DATA )                    017450   SP-HEAD236
  290 FORMAT (1H*,16H TARGET HEADING=,F4.0,8H DEGREES,10X,1H*)          017460   SP-HEAD237
  300 FORMAT (1H*,23H TARGET'S REFLECTIVITY=,F4.2,11X,1H*)              017470   SP-HEAD238
  310 FORMAT (1H*,26H NOMINAL GUN-TARGET RANGE=,F4.0,3H KM,5X,1H*)      017480   SP-HEAD239
  320 FORMAT (1H*,9H ANGLE T=,F4.0,8H DEGREES,17X,1H*)                  017490   SP-HEAD240
  330 FORMAT (1H*,17H DEFLECTION BIAS=,F5.0,7H METERS,9X,1H*)           017500   SP-HEAD241
  340 FORMAT (1H*,20H SEEKER SENSITIVITY=,F4.0,12H JOULES/M**2,2X,1H*)  017510   SP-HEAD242
  350 FORMAT (1H*,26H PROB SMOKE KILLS MISSION=,F4.2,8X,1H*)            017520   SP-HEAD243
  360 FORMAT (32H TEMPORARY OPTION IN EFFECT FOR ,A6)                   017530   SP-HEAD244
  370 FORMAT (15H RECORD NAMED: ,A10,22H USED TO OVERRIDE FOR ,A6)      017540   SP-HEAD245
  380 FORMAT (A4,6X)                                                    017550   SP-HEAD246
  390 FORMAT (16H REDATA FOR: TR=,F4.0,4H DT=,I1,5H VEL=,F2.0,5H GTR=,F3017560   SP-HEAD247
     1.0,6H REFL=,F4.2,8H ANGLET=,F4.0,6H DEFR=,F4.0,7H TGTHD=,F3.0,7H S017570   SP-HEAD248
     2KSEN=,F3.0)                                                       017580   SP-HEAD249
  400 FORMAT (1H1,5(10H* * * * * ),2X,21HC A S E   N U M B E R,I5,2X,5(1017590   SP-HEAD250
     10H* * * * * ),/)                                                  017600   SP-HEAD251
  410 FORMAT (1H ,1X,23HNUMBER OF REPLICATIONS=,I5,/)                   017610   SP-HEAD252
  420 FORMAT (8(5H*****))                                               017620   SP-HEAD253
  430 FORMAT (1H*,38X,1H*)                                              017630   SP-HEAD254
  440 FORMAT (1H*,16X,6HTARGET,16X,1H*)                                 017640   SP-HEAD255
  450 FORMAT (1H*,6H TYPE ,I2,30X,1H*)                                  017650   SP-HEAD256
  460 FORMAT (1H*,20H NUMBER OF VEHICLES=,I2,16X,1H*)                   017660   SP-HEAD257
  470 FORMAT (1H*,32H MEAN DISTANCE BETWEEN VEHICLES=,F4.0,3HM *)       017670   SP-HEAD258
  480 FORMAT (1H*,17H TARGET VELOCITY=,F2.0,6H M/SEC,13X,1H*)           017680   SP-HEAD259
  490 FORMAT (1H*,13X,11HFIRING INFO,14X,1H*)                           017690   SP-HEAD260
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 43 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

  500 FORMAT (1H*,33H NUMBER OF ROUNDS PER ENGAGEMENT=,I2,3X,1H*)        017700  SP-HEAD261
  510 FORMAT (1H*,19H ROUND RELIABILITY=,F4.2,15X,1H*)                    017710  SP-HEAD262
  520 FORMAT (1H*,21H TIME BETWEEN ROUNDS=,F4.0,4H SEC,9X,1H*)            017720  SP-HEAD263
  530 FORMAT (1H*,16H TIME OF FLIGHT=,F5.1,4H SEC,13X,1H*)                017730  SP-HEAD264
  540 FORMAT (1H*,17X,4HMISC,17X,1H*)                                     017740  SP-HEAD265
  550 FORMAT (1H*,26H DELTA T3 NOT ASSUMED ZERO,12X,1H*)                  017750  SP-HEAD266
  560 FORMAT (1H*,6H RUNC=,F5.0,1HM,26X,1H*)                              017760  SP-HEAD267
  570 FORMAT (1H*,28H DELTA T3 ASSUMED TO BE ZERO,10X,1H*)                017770  SP-HEAD268
  580 FORMAT (1H*,4X,29HDETECT, COMMO, AND PROC TIMES,5X,1H*)             017780  SP-HEAD269
  590 FORMAT (1H*,29H PARAMETERIZED RESPONSE TIME=,F4.0,6H SEC *)         017790  SP-HEAD270
  600 FORMAT (1H*,1X,3A10,7X,1H*)                                         017800  SP-HEAD271
  610 FORMAT (1H*,1X,2A10,17X,1H*)                                        017810  SP-HEAD272
  620 FORMAT (1H*,1X,A10,27X,1H*)                                         017820  SP-HEAD273
  630 FORMAT (1H*,23H NOMINAL RESPONSE TIME=,F4.0,4H SEC,7X,1H*)          017830  SP-HEAD274
  640 FORMAT (1H*,30H PROB SUCCESSFUL DIGITAL TRAN=,F5.3,3X,1H*)          017840  SP-HEAD275
  650 FORMAT (1H*,28H PROB SUCCESSFUL VOICE TRAN=,F5.3,5X,1H*)            017850  SP-HEAD276
  660 FORMAT (1H*,14X,10HDESIGNATOR,14X,1H*)                              017860  SP-HEAD277
  670 FORMAT (1H*,6H TYPE ,I2,30X,1H*)                                    017870  SP-HEAD278
  680 FORMAT (1H*,26H MAXIMUM DESIGNATOR RANGE=,F5.0,2H M,5X,1H*)         017880  SP-HEAD279
  690 FORMAT (1H*,16H FAIL-OUT RANGE=,F5.0,2H M,15X,1H*)                  017890  SP-HEAD280
  700 FORMAT (1H*,1X,3A10,7X,1H*)                                         017900  SP-HEAD281
  710 FORMAT (1H*,8X,21HDESIGNATOR SUPPRESSED,9X,1H*)                     017910  SP-HEAD282
  720 FORMAT (1H*,34H PROB SUPPRESSED BY INDIRECT FIRE=,F3.2,2H *)        017920  SP-HEAD283
  730 FORMAT (6X,I2,2X)                                                   017930  SP-HEAD284
  740 FORMAT (1H*,32H PROB SUPPRESSED BY DIRECT FIRE:,1X,A4,2H *)         017940  SP-HEAD285
  750 FORMAT (1H*,16X,7HWEATHER,15X,1H*)                                  017950  SP-HEAD286
  760 FORMAT (A3,A4,3X)                                                   017960  SP-HEAD287
  770 FORMAT (1H*,10H MONTH OF ,A3,25X,1H*)                               017970  SP-HEAD288
  780 FORMAT (1H*,14H TIME OF DAY: ,A4,20X,1H*)                           017980  SP-HEAD289
  790 FORMAT (1H*,27H OVERRIDE WITH WEATHER REC ,A10,2H *)                017990  SP-HEAD290
  800 FORMAT (1H*,32H USING "TEMPORARY" WEATHER DATA ,6X,1H*)             018000  SP-HEAD291
  810 FORMAT (1H*,11X,15HTERRAIN AND LOS,12X,1H*)                         018010  SP-HEAD292
  820 FORMAT (6X,I2,2X)                                                   018020  SP-HEAD293
  830 FORMAT (1H*,1X,2A10,17X,1H*)                                        018030  SP-HEAD294
  840 FORMAT (1H*,27H OVERRIDE WITH TERRAIN REC ,A10,2H *)                018040  SP-HEAD295
  850 FORMAT (1H*,31H USING "TEMPORARY" TERRAIN DATA,7X,1H*)              018050  SP-HEAD296
  860 FORMAT (1H*,1X,3A10,A7,1H*)                                         018060  SP-HEAD297
  870 FORMAT (1H*,14X,10HOBSCURANTS,14X,1H*)                              018070  SP-HEAD298
  880 FORMAT (1H*,24H VOLUME OF SMOKE FIRED: ,I4,3H * ,I4,3X,1H*)         018080  SP-HEAD299
  890 FORMAT (1H*,25H PROB DUST KILLS MISSION=,F4.2,9X,1H*)               018090  SP-HEAD300
  900 FORMAT (1H ,2X,4A10,3X,4A10,3X,4A10)                                018100  SP-HEAD301
  910 FORMAT (/)                                                          018110  SP-HEAD302
  920 FORMAT (1H ,6(8H$$   $$  ),8X,15HC U M M E N T S,9X,6(8H  $$   $$),/018120  SP-HEAD303
     1)                                                                   018130  SP-HEAD304
  930 FORMAT (1H ,13A10)                                                  018140  SP-HEAD305
      END                                                                 018150  SP-HEAD306

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPL PROGRAM - CONT'D

(PAGE 44 OF 32)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      SUBROUTINE HITCHK                                                 018170  SP-HITCHK2
C                                                                       018180  SP-HITCHK3
      COMMON /HIT/ INDEX(6,5,2), PETBL(60,10,7), DLTT(6), PKTBL(10,2),  000410  CM+HIT   2
     1  NRNGTT, RNGTTF(20), TTF(3,20,2), NFNGPS, PNGPST(10), PSTTBL(10, 000420  CM+HIT   3
     2  3)                                                              000430  CM+HIT   4
      COMMON /KTEST/ KTEST(20,6)                                        000490  CM+KTEST 2
      COMMON /RANDOM/ IR1, IR2, IR3, IR4, IR5, IR6, IR7, IR8, IR9, IR10,000640  CM+RANDOM2
     1  IR11, IR12, IR13, IR14, IR15, IR16, IR17, IR18, IR19, IR20, IR21000650  CM+RANDOM3
     2  , IR22, IR23, IR24, IR25, IR26, IR27, IR28                      000660  CM+RANDOM4
      COMMON /RANGE/ DTRNG, MXDRNG, VISLIM, RNGNOW                      000680  CM+RANGE 2
      REAL MXDRNG                                                       000690  CM+RANGE 3
      COMMON /RPLCTN/ KREP, NREP, KRF, NRF, NVEHKL                      000770  CM+RPLCTN2
      COMMON /RSPTIM/ RATRTM(3), XMVADF(2), DUM1, RCSPTM(2,2), XMTTIM(2,000790  CM+RSPTIM2
     1  2), TMEAN(2,2), TSIGMA(2,2), DETTMA(10,2), NDT, TRAPPY(3,2),    000800  CM+RSPTIM3
     2  PRSPTM, TPBAR, PRSDT, PRSVT, ALPHA(2,2), BETA(2,2), DETTIM,     000810  CM+RSPTIM4
     3  IDSGTP, RUNC, PCPMSG, PRDOW                                     000820  CM+RSPTIM5
      COMMON /TARGET/ TVEL, NVEHCL, DISRVH, ITGTTP, IMUTS, ITGTPS       001000  CM+TARGET2
      COMMON /TIME/ DURLDS, TIMNOW, TOF, TRP, TIMPA, TIMPA1             001020  CM+TIME  2
C                                                                       018270  SP-HITCH12
      DO 100 IK=1,NRNGTT                                                018280  SP-HITCH13
      IF (RNGNOW.LT.RNGTTF(IK).OR.RNGTTF(IK+1).LE.RNGNOW) GO TO 100     018290  SP-HITCH14
      IPM=IK                                                            018300  SP-HITCH15
      GO TO 110                                                         018310  SP-HITCH16
  100 CONTINUE                                                          018320  SP-HITCH17
      STOP ' IN HITCHK: FAILED TO BRACKET RANGE '                       018330  SP-HITCH18
C                                                                       018340  SP-HITCH19
  110 TFRAC1=(RNGNOW-RNGTTF(IPM))/(RNGTTF(IPM+1)-RNGTTF(IPM))           018350  SP-HITCH20
      PH=TTF(IDSGTP,IPM,ITGTPS)+TFRAC1*(TTF(IDSGTP,IPM+1,ITGTPS)        018360  SP-HITCH21
     1  -TTF(IDSGTP,IPM,ITGTPS))                                        018370  SP-HITCH22
C                                                                       018380  SP-HITCH23
      RN20=URAN31(IR20)                                                 018390  SP-HITCH24
      KTEST(18,KRF)=KTEST(18,KRF)+1                                     018400  SP-HITCH25
      IF (RN20.LE.PH) GO TO 120                                         018410  SP-HITCH26
C                                                                       018420  SP-HITCH27
      IABRT=18                                                          018430  SP-HITCH28
      CALL ABRTTL (IABRT)                                               018440  SP-HITCH29
C                                                                       018450  SP-HITCH30
  120 RETURN                                                            018460  SP-HITCH31
      END                                                               018470  SP-HITCH32

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 45 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      SUBROUTINE INITLZ                                                     018490   SR-INITLZ2
C                                                                           018500   SR-INITLZ3
      COMMON /ABORT/ NABORT(20,6), NSABLM                                   000110   CM+ABORT 2
      COMMON /ACHAR/ CHAR(10,120), NREC                                     000150   CM+ACHAR 2
      COMMON /DVALUE/ D(70), DPPT, DDT, DIDN, DIDIGT, DSMK(2)               000280   CM+DVALUE2
      COMMON /FLAG/ ENDREP, IDN, IPPEPL, IDIGTL, IPPTY, IPNDOC, IT3FLG,     000300   CM+FLAG  2
     1  ENDFLG                                                              000310   CM+FLAG  3
      LOGICAL ENDFLG, ENDREP                                                000320   CM+FLAG  4
      COMMON /ICODE/ ICODE                                                  000450   CM+ICODE 2
      COMMON /INDEX1/ INDX11(2001)                                          000470   CM+INDEX12
      COMMON /RUNDAT/ DAT, TIM, NCASE, ICCPFG, IPPFG                        000840   CM+RUNDAT2
      COMMON /UVALUE/ U(70), UPPT, UDT, UIDN, UIDIGT, USMK(2)               001040   CM+UVALUE2
      DIMENSION IRAY(6)                                                     018590   SP-INITL12
      DIMENSION IOPTNM(70)                                                  018600   SP-INITL13
C                                                                           018610   SP-INITL14
      EXTERNAL NOREC                                                        018620   SP-INITL15
C                                                                           018630   SP-INITL16
      EQUIVALENCE (ICODE,CODE)                                              018640   SR-INITL17
C                                                                           018650   SR-INITL18
      IRAY(1)=0                                                             018660   SP-INITL19
      IRAY(5)=LOCF(NOREC)                                                   018670   SP-INITL20
      CALL SYSTEMC (104,IRAY)                                               018680   SR-INITL21
C                                                                           018690   SP-INITL22
      CALL OPENMS (11,INDX11,2001,1)                                        018700   SP-INITL23
C                                                                           018710   SP-INITL24
      ICODE=8HOPTNNUMS                                                      018720   SP-INITL25
      CALL READMS (11,IOPTNM,70,ICODE)                                      018730   SR-INITL26
C                                                                           018740   SP-INITL27
      DO 100 I=1,NREC                                                       018750   SR-INITL28
      J=CHAR(1,I)                                                           018760   SP-INITL29
      CHAR(2,I)=IOPTNM(J)                                                   018770   SR-INITL30
  100 CONTINUE                                                              018780   SP-INITL31
C                                                                           018790   SP-INITL32
      CALL DFAULT                                                           018800   SP-INITL33
      CALL USET                                                             018810   SR-INITL34
C                                                                           018820   SP-INITL35
      NSABLM=11                                                             018830   SR-INITL36
      ENDFLG=.FALSE.                                                        018840   SP-INITL37
C                                                                           018850   SP-INITL38
      RETURN                                                                018860   SP-INITL39
      END                                                                   018870   SP-INITL40

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 46 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE INPUT                                                  018890  SP-INPUT 2
C                                                                       018900  SP-INPUT 3
      COMMON /ABORT/ MABORT(20,6), NSABLM                               000110  CM+ABORT 2
      COMMON /BAIL/ BLDPNG, IBAIL                                       000170  CM+BAIL  2
      COMMON /COMENT/ CMENT(13,20), COMBUF(13), ICMNT, NTPC             000190  CM+COMENT2
      COMMON /DESPNG/ DESPNG(3,2)                                       000210  CM+DESPNG2
      COMMON /DISPLY/ DISPT(24), DISPM(12), PECLBL(8), TMPLBL(8)        000230  CM+DISPLY2
      COMMON /DODF/ DEDOKL(10,3), NDESP, NDOKIL, DOKILD                 000250  CM+DODF  2
      LOGICAL DOKILD                                                    000260  CM+DODF  3
      COMMON /DVALUE/ D(70), DPPT, DDT, DIDN, DIDIGT, DSMK(2)           000280  CM+DVALUE2
      COMMON /FLAG/ ENDREP, IDN, IREEPL, IDIGTL, IPRTY, IPNDOC, IT3FLG, 000300  CM+FLAG  2
     1 ENDFLG                                                           000310  CM+FLAG  3
      LOGICAL ENDFLG, ENDREP                                            000320  CM+FLAG  4
      COMMON /FLTTIM/ TOFARY(16)                                        000340  CM+FLTTIM2
      COMMON /HIT/ INDEX(6,5,2), PETBL(60,10,7), DLTT(6), PKTBL(10,2),  000410  CM+HIT   2
     1 NPNGTT, PNGTTF(20), TTF(3,20,2), NPNGPS, PNGPST(10), PSTTBL(10,  000420  CM+HIT   3
     2 3)                                                               000430  CM+HIT   4
      COMMON /ICODE/ ICODE                                              000450  CM+ICODE 2
      COMMON /INDEX1/ INDX11(2001)                                      000470  CM+INDEX12
      COMMON /LOGFLG/ FIPSTL, SEQNML, SECL(70), SUPTEC, FTSILL          000510  CM+LOGFLG2
      LOGICAL FIPSTL, SEQNML, SPCL, SUPTEC, FTSILL                      000520  CM+LOGFLG3
      COMMON /MISC/ DOAPPK, PDSTPL, PELIF                               000540  CM+MISC  2
      COMMON /OVER/ IOVER(70)                                           000560  CM+OVER  2
      COMMON /PEDESC/ ANGLET, GTPNG, PEFL, PEFB, TGTHD, SKSEN           000580  CM+PEDESC2
      COMMON /POINT/ IPOINT(70)                                         000600  CM+POINT 2
      COMMON /RANDOC/ TCPIT, NPLOSP, PNGPLS(10), PPRLOS(10)             000620  CM+RANDOC2
      COMMON /RANDOM/ IP1, IP2, IP3, IR4, IP5, IP6, IP7, IP8, IP9, IP10,000640  CM+RANDOM2
     1 IP11, IP12, IP13, IP14, IP15, IP16, IP17, IP18, IP19, IP20, IP21000650  CM+RANDOM3
     2 , IP22, IP23, IP24, IP25, IP26, IP27, IP28                       000660  CM+RANDOM4
      COMMON /RANGE/ DTENG, MXDPNG, VISLIM, PNGNOW                      000680  CM+RANGE 2
      REAL MXDENG                                                       000690  CM+RANGE 3
      COMMON /RECNAM/ WCODE, ACCODE, PSCODE, DECODE, ROCODE, PECODE,    000710  CM+RECNAM2
     1 PSCODE, AICODE                                                   000720  CM+RECNAM3
      COMMON /RNGLOS/ NPP, NPD, NPNGCL, CPNGD(11,2), SEGLOS(11,11),      000740  CM+RNGLOS2
     1 PNGCLB(11), VELTBL(2,2)                                          000750  CM+RNGLOS3
      COMMON /RPLCTN/ KPEP, NPEP, KFF, NRF, NVEHKL                      000770  CM+RPLCTN2
      COMMON /RSPTIM/ RATRIM(3), XMVADF(2), DUM1, RCSPTM(2,2), XMTTIM(2,000790  CM+RSPTIM2
     1 2), TMEAN(2,2), TSIGMA(2,2), DETTMA(10,2), HDT, IPAPPY(3,2),     000800  CM+RSPTIM3
     2 PESPIM, TRBAP, PESDT, PESVT, ALPHA(2,2), BETA(2,2), DETTIM,      000810  CM+RSPTIM4
     3 IDSGTP, RUMC, PCPUSG, PRDOW                                      000820  CM+RSPTIM5
      COMMON /RUNDAT/ DAT, TIM, NCASE, ICCPEG, IPPEG                    000840  CM+RUNDAT2
      COMMON /SMOKED/ SMK2(3,3,2), SMK5(3,3,2), FRONT, NSMK2, NSMK5,    000920  CM+SMOKED2
     1 PSMKPL                                                           000930  CM+SMOKED3
      COMMON /SYMBOL/ BLANK, DECPNT, ALFPET(27), ANUMBP(11), SEP(5),    000970  CM+SYMBOL2
     1 DOLLAP                                                           000980  CM+SYMBOL3
      COMMON /TARGET/ TVEL, NVEHCL, DISRVH, ITGTTP, IMUTS, ITGTPS       001000  CM+TARGET2
      COMMON /TIME/ DUELOS, TIMNOW, TOF, TPP, TIMPA, TIMPA1             001020  CM+TIME  2
      COMMON /UVALUE/ U(70), UPPT, UDT, UIDN, UIDIGT, USMK(2)           001040  CM+UVALUE2
      COMMON /WEATHR/ PPCLCG, PPCLLS(2), PPGCEL(11,2), W(6,11,2),       001060  CM+WEATHR2
     1 VIS(11), CLOUD(6), HCC, HVL, PAPQL(6), WNDSPD(3,3), HUMID(2),    001070  CM+WEATHR3
     2 WSUM(2), WSFSUM(2), ICC, IVL                                     001080  CM+WEATHR4
      EXTERNAL NOFFC                                                    001120  DM+L     2
C                                                                       001130  DM+L     3
      DIMENSION IPAY(6)                                                 001140  DM+L     4

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 47 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)

      DIMENSION A(80), B(8,10), IT(10)                                           001150  DM+L     5
      DIMENSION NCRD(70), CH(10)                                                 001160  DM+L     6
      DIMENSION UVALUE(70), DVALUE(70)                                           001170  DM+L     7
C                                                                                001180  DM+L     8
      EQUIVALENCE (ICODE,CODE)                                                   001190  DM+L     9
      EQUIVALENCE (U(1),UVALUE(1)), (D(1),DVALUE(1))                             001200  DM+L    10
C                                                                                001220  DM+M     2
      DIMENSION WDATA(193), ACODAT(161), PSPDAT(49), DFSDAT(31)                  001230  DM+M     3
      DIMENSION RODATA(22), PEDATA(4260), PSTDAT(41), AINVDA(167)                001240  DM+M     4
C                                                                                001250  DM+M     5
      EQUIVALENCE (WDATA(1),PRCLCC)                                              001260  DM+M     6
      EQUIVALENCE (ACODAT(1),NPP)                                                001270  DM+M     7
      EQUIVALENCE (PSPDAT(1),BATPTM(1))                                          001280  DM+M     8
      EQUIVALENCE (DFSDAT(1),DFDOKL(1))                                          001290  DM+M     9
      EQUIVALENCE (RODATA(1),TCPIT)                                              001300  DM+M    10
      EQUIVALENCE (PEDATA(1),INDEX(1))                                           001310  DM+M    11
      EQUIVALENCE (PSTDAT(1),NPNGPS)                                             001320  DM+M    12
      EQUIVALENCE (AINVDA(1),DLTT(1))                                            001330  DM+M    13
C                                                                                019240  SP-INPUT37
      IPAY(1)=0                                                                  019250  SP-INPUT38
      IPAY(5)=LOCF(NDREC)                                                        019260  SP-INPUT39
      CALL SYSTEMC (104,IPAY)                                                    019270  SP-INPUT40
C                                                                                019280  SP-INPUT41
      DO 110 I=1,13                                                              019290  SP-INPUT42
      COMBUF(I)=BLANK                                                            019300  SP-INPUT43
      DO 100 J=1,20                                                              019310  SP-INPUT44
  100 CMENT(I,J)=BLANK                                                           019320  SP-INPUT45
  110 CONTINUE                                                                   019330  SP-INPUT46
      DO 120 I=1,70                                                              019340  SP-INPUT47
  120 NCRD(I)=0                                                                  019350  SP-INPUT48
C                                                                                019360  SP-INPUT49
      ICMNT=0                                                                    019370  SP-INPUT50
      NLPTC=0                                                                    019380  SP-INPUT51
      NTPC=0                                                                     019390  SP-INPUT52
      NCASE=NCASE+1                                                              019400  SP-INPUT53
C                                                                                019410  SP-INPUT54
      WRITE (8,770) NCASE                                                        019420  SP-INPUT55
C                                                                                019430  SP-INPUT56
C   DO "READ-A-LINE-OF-INPUT" UNTIL "END","ENDE",OR EOF ENCOUNTERED              019440  SP-INPUT57
C                                                                                019450  SP-INPUT58
  130 READ (5,870) (A(I),I=1,80)                                                 019460  SP-INPUT59
      IF (EOF(5).EQ.0.0) GO TO 140                                               019470  SP-INPUT60
      ENDFLG=.TRUE.                                                              019480  SP-INPUT61
      IF (NLPTC.EQ.0) STOP ' IN INPUT: EOF ENCOUNTERED '                         019490  SP-INPUT62
      ENCODE (100,890,CH(1) )                                                    019500  SP-INPUT63
      CALL COMMNT (CH,10,.FALSE.,.FALSE.)                                        019510  SP-INPUT64
      GO TO 490                                                                  019520  SP-INPUT65
C                                                                                019530  SP-INPUT66
  140 WRITE (8,880) (A(I),I=1,80)                                                019540  SP-INPUT67
C                                                                                019550  SP-INPUT68
      IF (A(1).NE.1H$) GO TO 150                                                 019560  SP-INPUT69
      CALL COMMNT (A,80,.TRUE.,.FALSE.)                                          019570  SP-INPUT70
      GO TO 130                                                                  019580  SP-INPUT71
C                                                                                019590  SP-INPUT72

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)
```

LISTING D-1  FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 48 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901 (READ VERTICALLY)

  150 NLRTC=NLRTC+1                                                          019600   SP-INPUT73
      CALL SEPREC (A,80,B,8,10)                                              019610   SP-INPUT74
      CALL FINDIT (B,8,10,IT,10)                                             019620   SP-INPUT75
      IT1=IT(1)                                                              019630   SP-INPUT76
      IGO=IPOINT(IT1)                                                        019640   SP-INPUT77
      ICRD=IT1                                                               019650   SP-INPUT78
      IF (6.LE.IT1.AND.IT1.LE.8) ICRD=IT(2)                                  019660   SP-INPUT79
      NCRD(ICRD)=NCRD(ICRD)+1                                                019670   SP-INPUT80
      IF (ICRD.EQ.1.AND.NCASE.GT.1) STOP ' IN INPUT: FIRST CASE CARD IN      019680   SP-INPUT81
     1SUBSEQUENT CASE '                                                      019690   SP-INPUT82
      IF (ICRD.EQ.2.AND.(NCRD(1).GT.0.OR.NCRD(2).GT.1)) STOP ' IN INPUT:019700   SP-INPUT83
     1 EXTRA NEXT CASE CARD '                                                019710   SP-INPUT84
      IF (NCRD(ICRD).EQ.1) GO TO 160                                         019720   SP-INPUT85
      ENCODE (100,810,CM(1) )(A(I),I=1,80)                                   019730   SP-INPUT86
      CALL COMMNT (CM,10,.FALSE.,.FALSE.)                                    019740   SP-INPUT87
      ENCODE (100,820,CM(1) )                                                019750   SP-INPUT88
      CALL COMMNT (CM,10,.FALSE.,.FALSE.)                                    019760   SP-INPUT89
C                                                                            019770   SP-INPUT90
  160 GO TO (130,190,490,170,180), IGO                                       019780   SP-INPUT91
C                                                                            019790   SP-INPUT92
  170 ENDFLG=.TRUE.                                                          019800   SP-INPUT93
      GO TO 490                                                              019810   SP-INPUT94
C                                                                            019820   SP-INPUT95
  180 IF (NLRTC.NE.1) GO TO 200                                              019830   SP-INPUT96
      ENCODE (100,900,CM(1) )                                                019840   SP-INPUT97
      CALL COMMNT (CM,10,.FALSE.,.FALSE.)                                    019850   SP-INPUT98
  190 IF (IGO.EQ.2.AND.IT(2).EQ.2) GO TO 130                                 019860   SP-INPUT99
      CALL USET                                                              019870   SP-INPU100
      IF (IGO.EQ.2) GO TO 130                                                019880   SP-INPU101
C                                                                            019890   SP-INPU102
  200 CONTINUE                                                               019900   SP-INPU103
C                                                                            019910   SP-INPU104
C  CASE IT1 OF "KEYWORD"                                                     019920   SP-INPU105
C                                                                            019930   SP-INPU106
      GO TO (210,210,210,210,220,240,250,260,230,210,270,280,280,280,280019940   SP-INPU107
     1  ,280,290,210,210,210,300,310,320,310,210,210,310,310,210,210,340019950   SP-INPU108
     2  ,380,210,210,210,210,210,210,210,210,360,420,420,360,450,360,360019960   SP-INPU109
     3  ,360,360,360,360,360,360,360,360,360,360,360,360,360,360,360,210019970   SP-INPU110
     4  ,210,210,210,210,210,210,210,210), IT1                               019980   SP-INPU111
C                                                                            019990   SP-INPU112
  210 WRITE (6,790) (A(I),I=1,80),IT1                                        020000   SP-INPU113
      STOP ' IN INPUT: ERROR NUMBER 1 '                                      020010   SP-INPU114
C                                                                            020020   SP-INPU115
  220 IF (IT(2).EQ.1) IPREG=IT(3)-1                                          020030   SP-INPU116
      IF (IT(2).EQ.2) ICPREG=IT(3)-1                                         020040   SP-INPU117
      GO TO 480                                                              020050   SP-INPU118
C                                                                            020060   SP-INPU119
  230 SHPTEC=.TRUE.                                                          020070   SP-INPU120
      IF (IT(3).EQ.2) SHPTEC=.FALSE.                                         020080   SP-INPU121
      GO TO 480                                                              020090   SP-INPU122
C                                                                            020100   SP-INPU123
  240 IT2=IT(2)                                                              020110   SP-INPU124
      UVALUE(IT2)=B(1,2)                                                     020120   SP-INPU125
      IOVER(IT2)=1                                                           020130   SP-INPU126

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 49 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      GO TO 480                                                        020140   SP-INPU127
C                                                                      020150   SP-INPU128
  250 IT2=IT(2)                                                        020160   SP-INPU129
      IOVER(IT2)=2                                                     020170   SP-INPU130
      CALL CREAD (IT2)                                                 020180   SP-INPU131
      GO TO 480                                                        020190   SP-INPU132
C                                                                      020200   SP-INPU133
  260 IT2=IT(2)                                                        020210   SP-INPU134
      IOVER(IT2)=0                                                     020220   SP-INPU135
      IF (NCPD(IT2).EQ.0) U(IT2)=D(IT2)                                020230   SP-INPU136
      GO TO 480                                                        020240   SP-INPU137
C                                                                      020250   SP-INPU138
  270 UVALUE(11)=IT(2)                                                 020260   SP-INPU139
      GO TO 480                                                        020270   SP-INPU140
C                                                                      020280   SP-INPU141
  280 UVALUE(IT1)=IT(2)                                                020290   SP-INPU142
      GO TO 480                                                        020300   SP-INPU143
C                                                                      020310   SP-INPU144
  290 ENCODE (10,800,UVALUE(27) )RECLBL(7),IT(2)                       020320   SP-INPU145
      GO TO 480                                                        020330   SP-INPU146
C                                                                      020340   SP-INPU147
  300 IT2=IT(2)                                                        020350   SP-INPU148
      IT3=IT(3)                                                        020360   SP-INPU149
      IOVER(IT1)=0                                                     020370   SP-INPU150
      ENCODE (10,840,UVALUE(21) )DISPN(IT2),DISPT(IT3),IT1             020380   SP-INPU151
      GO TO 480                                                        020390   SP-INPU152
C                                                                      020400   SP-INPU153
  310 ENCODE (10,850,UVALUE(IT1) )RECLBL(IT1-20),IT(2),IT1             020410   SP-INPU154
      IOVER(IT1)=0                                                     020420   SP-INPU155
      GO TO 480                                                        020430   SP-INPU156
C                                                                      020440   SP-INPU157
  320 UION=IT(4)                                                       020450   SP-INPU158
      IOVER(IT1)=0                                                     020460   SP-INPU159
      IF (IT(2).EQ.11) GO TO 330                                       020470   SP-INPU160
      ENCODE (10,850,UVALUE(IT1) )RECLBL(IT1-20),IT(2),IT1             020480   SP-INPU161
      UPRT=0.0                                                         020490   SP-INPU162
      UIDIGT=IT(3)                                                     020500   SP-INPU163
      FTSILL=.FALSE.                                                   020510   SP-INPU164
      IF (IT(2).EQ.1) FTSILL=.TRUE.                                    020520   SP-INPU165
      GO TO 480                                                        020530   SP-INPU166
  330 DECODE (10,910,R(1,3) )UPRT                                      020540   SP-INPU167
      IF (UPRT.LE.0.0) STOP ' IN INPUT: PARAMETERIZED RESPONSE TIME <=0.020550   SP-INPU168
     10 '                                                              020560   SP-INPU169
      FTSILL=.FALSE.                                                   020570   SP-INPU170
      GO TO 480                                                        020580   SP-INPU171
C                                                                      020590   SP-INPU172
  340 IF (IT(2).EQ.1) GO TO 350                                        020600   SP-INPU173
      UVALUE(31)=1.0                                                   020610   SP-INPU174
      DECODE (10,910,R(1,2) )RUNC                                      020620   SP-INPU175
      GO TO 480                                                        020630   SP-INPU176
  350 UVALUE(31)=DVALUE(31)                                            020640   SP-INPU177
      GO TO 480                                                        020650   SP-INPU178
C                                                                      020660   SP-INPU179
  360 IF (IT(2).NE.1) GO TO 370                                        020670   SP-INPU180

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 50 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      UVALUE(IT1)=DVALUE(IT1)                                             020680   SP-INPU181
      GO TO 480                                                           020690   SP-INPU182
  370 DECODE (10,910,B(1,2) )UVALUE(IT1)                                  020700   SR-INPU183
      GO TO 480                                                           020710   SR-INPU184
C                                                                         020720   SR-INPU185
  380 IF (IT(2).EQ.1) GO TO 400                                           020730   SP-INPU186
      SPCL(IT1)=.TRUE.                                                    020740   SP-INPU187
      IF (IT(3).NE.1) GO TO 390                                           020750   SP-INPU188
      UVALUE(IT1)=DVALUE(IT1)                                             020760   SP-INPU189
      GO TO 480                                                           020770   SP-INPU190
  390 DECODE (10,910,B(1,3) )UVALUE(IT1)                                  020780   SR-INPU191
      GO TO 480                                                           020790   SR-INPU192
  400 SPCL(IT1)=.FALSE.                                                   020800   SR-INPU193
      DO 410 JK=3,4                                                       020810   SP-INPU194
      IF (IT(JK).EQ.1) USMK(JK-2)=DSMK(JK-2)                              020820   SR-INPU195
      IF (IT(JK).NE.1) DECODE (10,910,B(1,JK) )USMK(JK-2)                 020830   SP-INPU196
  410 CONTINUE                                                            020840   SR-INPU197
      GO TO 480                                                           020850   SR-INPU198
C                                                                         020860   SR-INPU199
  420 IF (IT(2).EQ.1) GO TO 430                                           020870   SR-INPU200
      SPCL(IT1)=.TRUE.                                                    020880   SR-INPU201
      DECODE (10,910,B(1,2) )UVALUE(IT1)                                  020890   SP-INPU202
      GO TO 440                                                           020900   SP-INPU203
  430 SPCL(IT1)=.FALSE.                                                   020910   SP-INPU204
  440 CONTINUE                                                            020920   SP-INPU205
      GO TO 480                                                           020930   SP-INPU206
C                                                                         020940   SP-INPU207
  450 IF (IT(2).EQ.1) GO TO 460                                           020950   SR-INPU208
      SPCL(IT1)=.TRUE.                                                    020960   SR-INPU209
      DECODE (10,910,B(1,2) )UVALUE(IT1)                                  020970   SP-INPU210
      GO TO 470                                                           020980   SP-INPU211
  460 SPCL(IT1)=.FALSE.                                                   020990   SP-INPU212
  470 CONTINUE                                                            021000   SP-INPU213
C                                                                         021010   SP-INPU214
C  END OF "KEYWORD" CASES                                                 021020   SR-INPU215
C                                                                         021030   SR-INPU216
  480 CONTINUE                                                            021040   SR-INPU217
C                                                                         021050   SP-INPU218
C  END OF "READ-A-LINE-OF-INPUT" LOOP                                     021060   SR-INPU219
C                                                                         021070   SP-INPU220
      GO TO 130                                                           021080   SP-INPU221
C                                                                         021090   SP-INPU222
C  * * * * * * * * * * * * * * * * * * * * * * *                          021100   SP-INPU223
C                                                                         021110   SP-INPU224
C  FILL IN VALUES TO BE PASSED TO REST OF PROGRAM VIA COMMON.             021120   SR-INPU225
C                                                                         021130   SR-INPU226
  490 CODE=UVALUE(21)                                                     021140   SR-INPU227
      DECODE (10,780,ICODE) XWTHP                                         021150   SR-INPU228
      DO 500 I=1,12                                                       021160   SP-INPU229
      IF (XWTHP.NE.DISPM(I)) GO TO 500                                    021170   SP-INPU230
      IMNTH=I                                                             021180   SP-INPU231
      GO TO 510                                                           021190   SP-INPU232
  500 CONTINUE                                                            021200   SP-INPU233
  510 CONTINUE                                                            021210   SP-INPU234

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 51 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)

      IWTHPL=193                                                        021220   SP-INPU235
      IF (IOVER(21).EQ.2) GO TO 520                                     021230   SP-INPU236
      CALL READMS (11,WDATA,IWTHPL,ICODE)                               021240   SR-INPU237
      WCODE=CODE                                                        021250   SP-INPU238
  520 DO 530 I=1,2                                                      021260   SP-INPU239
      WSUM(I)=0.0                                                       021270   SP-INPU240
  530 WCFSUM(I)=0.0                                                     021280   SR-INPU241
      DO 550 I=1,NVL                                                    021290   SP-INPU242
      DO 550 J=1,2                                                      021300   SP-INPU243
      WCFSUM(J)=WCFSUM(J)+PPGCFL(I,J)                                   021310   SP-INPU244
      DO 540 K=1,NCC                                                    021320   SP-INPU245
  540 WSUM(J)=WSUM(J)+W(K,I,J)                                          021330   SP-INPU246
  550 CONTINUE                                                          021340   SP-INPU247
C                                                                       021350   SP-INPU248
      CODE=UVALUE(22)                                                   021360   SP-INPU249
      IACQRL=161                                                        021370   SP-INPU250
      IF (IOVER(22).EQ.2) GO TO 560                                     021380   SP-INPU251
      CALL READMS (11,ACQDAT,IACQRL,ICODE)                              021390   SP-INPU252
      ACCODE=CODE                                                       021400   SP-INPU253
C                                                                       021410   SR-INPU254
  560 GTRNG=UVALUE(56)                                                  021420   SP-INPU255
      IGTP=GTRNG+.5                                                     021430   SP-INPU256
      TOF=TOFARY(IGTP)                                                  021440   SP-INPU257
      IF (SPCL(45)) TOF=UVALUE(45)                                      021450   SP-INPU258
C                                                                       021460   SP-INPU259
      CODE=UVALUE(23)                                                   021470   SP-INPU260
      PRSPTM=UPRT                                                       021480   SP-INPU261
      IRSPRL=49                                                         021490   SP-INPU262
      IF (IOVER(23).EQ.2) GO TO 570                                     021500   SP-INPU263
      CALL READMS (11,RSPDAT,IRSPRL,ICODE)                              021510   SP-INPU264
      RSCODE=CODE                                                       021520   SP-INPU265
  570 IDN=UIDN                                                          021530   SR-INPU266
      IDIGTL=UIDIGT                                                     021540   SP-INPU267
      IPPTY=0                                                           021550   SP-INPU268
      UMC=UVALUE(12)                                                    021560   SP-INPU269
      IF (UMC.EQ.1.) IPPTY=2                                            021570   SR-INPU270
      IPREPL=2                                                          021580   SP-INPU271
      IF (UMC.LT.3.) IPREPL=1                                           021590   SR-INPU272
      TRBAR=TRARRY(IPREPL+IPPTY,IDIGTL)+TOF                             021600   SP-INPU273
      IF (PRSPTM.NE.0.0) TRBAR=PRSPTM+TOF                               021610   SP-INPU274
      IF (ITSILL) GO TO 590                                             021620   SP-INPU275
      IF (PRSPTM.GT.0.0) GO TO 590                                      021630   SP-INPU276
      DO 580 I=1,2                                                      021640   SP-INPU277
      DO 580 J=1,2                                                      021650   SP-INPU278
      ALPHA(I,J)=(TMEAN(I,J)/TSIGMA(I,J))**2                            021660   SP-INPU279
      BETA(I,J)=TSIGMA(I,J)**2/TMEAN(I,J)                               021670   SR-INPU280
  580 CONTINUE                                                          021680   SP-INPU281
  590 CONTINUE                                                          021690   SP-INPU282
C                                                                       021700   SP-INPU283
      CODE=UVALUE(24)                                                   021710   SP-INPU284
      IDFSRL=31                                                         021720   SP-INPU285
      IF (IOVER(24).EQ.2) GO TO 600                                     021730   SP-INPU286
      CALL READMS (11,DFSDAT,IDFSRL,ICODE)                              021740   SR-INPU287
      DFCODE=CODE                                                       021750   SP-INPU288

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 52 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

  600 CONTINUE                                                              021760   SP-INPU289
C                                                                           021770   SP-INPU290
      IF (.NOT.SPCL(42)) GO TO 610                                          021780   SP-INPU291
      TVEL=UVALUE(42)                                                       021790   SP-INPU292
      GO TO 620                                                             021800   SP-INPU293
  610 IXXX=3.0-ABS(2.5-FLOAT(IMNTH))                                        021810   SP-INPU294
      TVEL=VELTBL(IDN,IXXX)                                                 021820   SR-INPU295
  620 CONTINUE                                                              021830   SR-INPU296
C                                                                           021840   SR-INPU297
      IDSGTP=UVALUE(16)                                                     021850   SP-INPU298
      IF (.NOT.SPCL(43)) GO TO 630                                          021860   SR-INPU299
      MXDRNG=UVALUE(43)                                                     021870   SR-INPU300
      GO TO 640                                                             021880   SP-INPU301
  630 MXDRNG=DESRNG(IDSGTP,IDN)                                             021890   SP-INPU302
  640 CONTINUE                                                              021900   SP-INPU303
C                                                                           021910   SR-INPU304
      IPNDUC=UVALUE(11)-1.0                                                 021920   SP-INPU305
      IF (IRNDOC.EQ.0.OR.IOVER(25).EQ.2) GO TO 670                          021930   SR-INPU306
      IF (IOVER(25).NE.1) GO TO 650                                         021940   SP-INPU307
      CODE=UVALUE(25)                                                       021950   SP-INPU308
      GO TO 660                                                             021960   SP-INPU309
  650 DECODE (10,830,H(22) )ITRPN                                           021970   SP-INPU310
      ITVEL=TVEL                                                            021980   SP-INPU311
      IF (ITVEL.EQ.9) ITVEL=8                                               021990   SP-INPU312
      IF (ITVEL.EQ.2) ITVEL=3                                               022000   SP-INPU313
      ENCODE (10,860,ICODE) RECLBL(5),ITVEL,ITPRN                           022010   SP-INPU314
  660 IPDRL=22                                                              022020   SR-INPU315
      CALL READMS (11,RDDATA,IPDRL,ICODE)                                   022030   SP-INPU316
      RDCODE=CODE                                                           022040   SP-INPU317
  670 CONTINUE                                                              022050   SR-INPU318
C                                                                           022060   SP-INPU319
      IF (IOVER(26).EQ.2) GO TO 700                                         022070   SR-INPU320
      IF (IOVER(26).NE.1) GO TO 680                                         022080   SR-INPU321
      CODE=UVALUE(26)                                                       022090   SR-INPU322
      GO TO 690                                                             022100   SP-INPU323
  ..0 ANGLFT=UVALUE(46)                                                     022110   SP-INPU324
       EFL=UVALUE(57)                                                       022120   SP-INPU325
        FB=UVALUE(58)                                                       022130   SP-INPU326
      ..THD=UVALUE(59)                                                      022140   SP-INPU327
      SKSEN=UVALUE(60)                                                      022150   SP-INPU328
           RENAME (TPRAP,IDSGTP,TVEL,GTRNG,REFL,ANGLFT,REFB,TGTHD,SKSEN,    022160   SR-INPU329
           \                                                                022170   SP-INPU330
  6.       ..4260                                                           022180   SP-INPU331
      ..LL READMS (11,REDATA,IPREL,ICODE)                                   022190   SR-INPU332
      RECODE=CODE                                                           022200   SP-INPU333
  700 CONTINUE                                                              022210   SR-INPU334
C                                                                           022220   SP-INPU335
      IF (IOVER(27).EQ.2) GO TO 710                                         022230   SR-INPU336
      CODE=UVALUE(27)                                                       022240   SP-INPU337
      ITPSPL=41                                                             022250   SF-INPU338
      CALL READMS (11,PSTRAT,ITPSPL,ICODE)                                  022260   SP-INPU339
      PSCODE=CODE                                                           022270   SP-INPU340
  710 CONTINUE                                                              022280   SP-INPU341
C                                                                           022290   SP-INPU342

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 53 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      IF (UVALUE(13).EQ.1.) GO TO 730                                       022300  SP-INPU343
      NRNGP1=NRNGPS                                                         022310  SP-INPU344
      DO 720 I=1,NRNGP1                                                     022320  SP-INPU345
      IF (PSTTBL(I,1).EQ.0.0) GO TO 720                                     022330  SR-INPU346
      PSTTBL(I,2)=PSTTBL(I,2)/(1.-PSTTBL(I,1))                              022340  SP-INPU347
      PSTTBL(I,3)=1.-PSTTBL(I,2)                                            022350  SR-INPU348
      PSTTBL(I,1)=0.0                                                       022360  SP-INPU349
  720 CONTINUE                                                              022370  SP-INPU350
  730 CONTINUE                                                              022380  SP-INPU351
C                                                                           022390  SR-INPU352
      IF (IDVER(28).EQ.2) GO TO 740                                         022400  SR-INPU353
      CODE=UVALUE(28)                                                       022410  SP-INPU354
      INVDPL=167                                                            022420  SP-INPU355
      CALL READMS (11,AINVDA,INVDPL,ICODE)                                  022430  SP-INPU356
      AICODE=CODE                                                           022440  SR-INPU357
  740 CONTINUE                                                              022450  SP-INPU358
C                                                                           022460  SR-INPU359
      ITGTTP=UVALUE(14)                                                     022470  SR-INPU360
C                                                                           022480  SR-INPU361
C                                                                           022490  SP-INPU362
      IT3FLG=UVALUE(31)                                                     022500  SR-INPU363
C                                                                           022510  SP-INPU364
      IF (SPCL(32)) GO TO 750                                               022520  SP-INPU365
      NSMK2=USMK(1)                                                         022530  SR-INPU366
      NSMK5=USMK(2)                                                         022540  SP-INPU367
      CALL PSMOKE                                                           022550  SP-INPU368
      GO TO 760                                                             022560  SP-INPU369
  750 PSMKKL=UVALUE(32)                                                     022570  SP-INPU370
  760 CONTINUE                                                              022580  SP-INPU371
C                                                                           022590  SP-INPU372
      DOARPK=UVALUE(41)                                                     022600  SP-INPU373
      BLORNG=UVALUE(44)                                                     022610  SP-INPU374
      PDSTKL=UVALUE(47)                                                     022620  SR-INPU375
      PRSVT=UVALUE(48)                                                      022630  SP-INPU376
      PRSDT=UVALUE(49)                                                      022640  SP-INPU377
      RELIF=UVALUE(50)                                                      022650  SP-INPU378
      TBR=UVALUE(51)                                                        022660  SP-INPU379
      NREP=UVALUE(52)                                                       022670  SR-INPU380
      NRF=UVALUE(53)                                                        022680  SR-INPU381
      NVEHCL=UVALUE(54)                                                     022690  SP-INPU382
      DISBVH=UVALUE(55)                                                     022700  SR-INPU383
      PCRMSG=UVALUE(61)                                                     022710  SP-INPU384
      PPDOW=UVALUE(62)                                                      022720  SP-INPU385
C                                                                           022730  SP-INPU386
C  IF CCPLOT AND/OR PPLOT ARE IMPLEMENTED AS REAL PLOT PROGRAMS RATHER      022740  SP-INPU387
C  THAN AS PROGRAM STUBS, THEN THE CALL STATEMENTS TO THEM (BELOW)          022750  SP-INPU388
C  SHOULD BE MOVED TO A SUBROUTINE THAT IS CALLED AT THE COMPLETION         022760  SP-INPU389
C  OF EACH REPLICATION (SUCH AS OUTPUT)                                     022770  SR-INPU390
C                                                                           022780  SP-INPU391
      IF (IPPFG.EQ.1) CALL PPLOT                                            022790  SP-INPU392
      IF (ICCPFG.EQ.1) CALL CCPLOT                                          022800  SR-INPU393
C                                                                           022810  SR-INPU394
      CALL HEADER                                                           022820  SP-INPU395
C                                                                           022830  SP-INPU396

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 54 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      CALL ECHO (8)                                                     022840  SR-INPU397
C                                                                       022850  SR-INPU398
      RETURN                                                            022860  SR-INPU399
C                                                                       022870  SR-INPU400
C                                                                       022880  SR-INPU401
C   * * *   F O R M A T    S T A T E M E N T S   * * *                  022890  SR-INPU402
C                                                                       022900  SR-INPU403
  770 FORMAT (1H ,50(1H*),1X,23HINPUT OPTIONS FOR CASE ,I3,1X,50(1H*))  022910  SR-INPU404
  780 FORMAT (A3,7X)                                                    022920  SR-INPU405
  790 FORMAT (1H ,23H ERROR IN INPUT. LINE: ,80A1,11H WITH IT1= ,I5,/,13022930  SR-INPU406
     1H IS IN ERROR )                                                   022940  SR-INPU407
  800 FORMAT (A6,I2.2,2H27)                                             022950  SR-INPU408
  810 FORMAT (1H ,18HYOUR INPUT LINE : ,1X,80A1)                        022960  SR-INPU409
  820 FORMAT (1H ,58HSUPERSEDES ONE OF YOUR PREVIOUS INPUT LINES FOR THI022970  SR-INPU410
     1S CASE )                                                          022980  SR-INPU411
  830 FORMAT (6X,I2,2X)                                                 022990  SR-INPU412
  840 FORMAT (A3,A4,I3.3)                                               023000  SR-INPU413
  850 FORMAT (A6,I2.2,I2.2)                                             023010  SR-INPU414
  860 FORMAT (A5,I1,3HTRN,I1)                                           023020  SR-INPU415
  870 FORMAT (80A1)                                                     023030  SR-INPU416
  880 FORMAT (1H ,80A1)                                                 023040  SR-INPU417
  890 FORMAT (1H ,71HNO ENDF CARD FOUND BUT ONE ASSUMED SINCE END OF INP023050  SR-INPU418
     1UT FILE ENCOUNTERED )                                             023060  SR-INPU419
  900 FORMAT (1H ,3HNO ,12H'FIRST CASE',4H OR ,11H'NEXT CASE',14H CARD F023070  SR-INPU420
     1OUND---,15H' NEXT CASE,DD',16H OPTION ASSUMED )                   023080  SR-INPU421
  910 FORMAT (F10.4)                                                    023090  SR-INPU422
      END                                                               023100  SR-INPU423

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 55 Of 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 9 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)

      SUBROUTINE LOSCHK                                              023120  SP-LOSCHK2
C                                                                    023130  SP-LOSCHK3
      COMMON /FLAG/ ENDREP, IDN, IPREPL, IDIGTL, IPRTY, IRNDOC, IT3FLG, 000300  CM*FLAG  2
     1  ENDFLG                                                       000310  CM*FLAG  3
      LOGICAL ENDFLG, ENDREP                                         000320  CM*FLAG  4
      COMMON /KTEST/ KTEST(20,6)                                     000490  CM*KTEST 2
      COMMON /RANDOC/ TCRIT, NPLOSP, RNGPLS(10), PPBLOS(10)          000620  CM*RANDOC2
      COMMON /RANDOM/ IR1, IR2, IR3, IR4, IR5, IR6, IR7, IR8, IR9, IR10,000640  CM*RANDOM2
     1  IR11, IR12, IR13, IR14, IR15, IR16, IR17, IR18, IR19, IR20, IR21000650  CM*RANDOM3
     2  , IR22, IR23, IR24, IR25, IR26, IR27, IR28                   000660  CM*RANDOM4
      COMMON /RANGE/ DTRNG, MXDRNG, VISLIM, RNGNOW                   000680  CM*RANGE 2
      REAL MXDRNG                                                    000690  CM*RANGE 3
      COMMON /RPLCTN/ KEEP, NREP, KRF, NRF, NVEHKL                   000770  CM*RPLCTN2
      COMMON /TARGET/ TVEL, NVEHCL, DISPVH, ITGTTP, IMUTS, ITGTPS    001000  CM*TARGET2
      COMMON /TIME/ DUPLOS, TIMNOW, TOF, TRP, TIMPA, TIMPAI          001020  CM*TIME  2
C                                                                    023220  SP-LOSCH12
      KTEST(12,KRF)=KTEST(12,KRF)+1                                  023230  SP-LOSCH13
      IF (IRNDOC.EQ.1) GO TO 100                                     023240  SP-LOSCH14
      IF (TIMPA.LE.DUPLOS) GO TO 130                                 023250  SP-LOSCH15
C                                                                    023260  SP-LOSCH16
      IABRT=12                                                       023270  SP-LOSCH17
      CALL ABRTTL (IABRT)                                            023280  SP-LOSCH18
C                                                                    023290  SP-LOSCH19
      GO TO 130                                                      023300  SP-LOSCH20
C                                                                    023310  SP-LOSCH21
  100 XNV=NVEHCL-NVEHKL                                              023320  SP-LOSCH22
      RNGNOW=AMAX1(0.0,DTRNG-TVEL*TIMPA)                             023330  SP-LOSCH23
C                                                                    023340  SP-LOSCH24
      DO 110 I=1,NPLOSP                                              023350  SP-LOSCH25
      IF (RNGNOW.LE.RNGPLS(I)) GO TO 120                             023360  SP-LOSCH26
  110 CONTINUE                                                       023370  SP-LOSCH27
C                                                                    023380  SP-LOSCH28
      STOP ' IN LOSCHK: ERROR WITH RANDOM OCCURRENCE OPTION '        023390  SP-LOSCH29
C                                                                    023400  SP-LOSCH30
  120 PPLOS=PPBLOS(I)                                                023410  SP-LOSCH31
      PRIV=1.-(1.-PPLOS)**XNV                                        023420  SP-LOSCH32
      RN18=URAN31(IR18)                                              023430  SP-LOSCH33
      IF (RN18.LE.PRIV) GO TO 130                                    023440  SP-LOSCH34
C                                                                    023450  SP-LOSCH35
      IABRT=12                                                       023460  SP-LOSCH36
      CALL ABRTTL (IABRT)                                            023470  SP-LOSCH37
C                                                                    023480  SP-LOSCH38
  130 RETURN                                                         023490  SP-LOSCH39
C                                                                    023500  SP-LOSCH40
      END                                                            023510  SP-LOSCH41

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888 9 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678 90 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 56 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE MINTRN                                                   023530   SP-MINTRN2
C                                                                         023540   SP-MINTRN3
      COMMON /HIT/ INDEX(4,5,2), PETBL(60,10,7), DLTT(6), PKTBL(10,2),    000410   CM+HIT   2
     1 NPNGTT, RNGTTF(20), TTF(3,20,2), NPNGPS, RNGPST(10), PSTTBL(10,    000420   CM+HIT   3
     2 3)                                                                 000430   CM+HIT   4
      COMMON /KTEST/ KTEST(20,6)                                          000490   CM+KTEST 2
      COMMON /RANDOM/ IP1, IP2, IP3, IP4, IP5, IP6, IP7, IP8, IP9, IP10,000640   CM+RANDOM2
     1 IP11, IP12, IP13, IP14, IP15, IP16, IP17, IP18, IP19, IP20, IP21000650   CM+RANDOM3
     2 , IP22, IR23, IP24, IP25, IP26, IP27, IP28                         000660   CM+RANDOM4
      COMMON /RANGE/ DTRNG, MXDRNG, VISLIM, RNGNOW                        000680   CM+RANGE 2
      REAL MXDRNG                                                         000690   CM+RANGE 3
      COMMON /RPLCTN/ KREP, NREP, KRF, NRF, NVEHKL                        000770   CM+RPLCTN2
      COMMON /TARGET/ TVEL, NVEHCL, DISRVH, ITGTTP, IMUTS, ITGTPS         001000   CM+TARGET2
      COMMON /TIME/ DURLOS, TIMNOW, TOF, TBP, TIMPA, TIMPA1               001020   CM+TIME  2
C                                                                         023620   SP-MINTR11
      DIMENSION PSTVAL(3)                                                 023630   SP-MINTR12
C                                                                         023640   SP-MINTR13
      RNGNOW=AMAX1(0.0,DTRNG-TVEL*TIMPA)                                  023650   SP-MINTR14
C                                                                         023660   SP-MINTR15
      DO 100 IJ=1,NPNGPS                                                  023670   SP-MINTR16
      IF (RNGNOW.LT.RNGPST(IJ).OR.RNGPST(IJ+1).LE.RNGNOW) GO TO 100       023680   SP-MINTR17
      IPN=IJ                                                              023690   SP-MINTR18
      GO TO 110                                                           023700   SP-MINTR19
  100 CONTINUE                                                            023710   SP-MINTR20
      STOP ' IN MINTRN: FAILED TO BRACKET RANGE '                         023720   SP-MINTR21
C                                                                         023730   SP-MINTR22
  110 TFRAC=(RNGNOW-RNGPST(IPN))/(RNGPST(IPN+1)-RNGPST(IPN))              023740   SP-MINTR23
C                                                                         023750   SP-MINTR24
      DO 120 J=1,3                                                        023760   SP-MINTR25
      PSTVAL(J)=PSTTBL(IPN,J)+TFRAC*(PSTTBL(IPN+1,J)-PSTTBL(IPN,J))       023770   SP-MINTR26
  120 CONTINUE                                                            023780   SP-MINTR27
C                                                                         023790   SP-MINTR28
      RN19=URAN31(IP19)                                                   023800   SP-MINTR29
      ITGTPS=0                                                            023810   SP-MINTR30
      IF (PSTVAL(1).LE.RN19) ITGTPS=1                                     023820   SP-MINTR31
      IF (PSTVAL(2)+PSTVAL(1).LE.RN19) ITGTPS=2                           023830   SP-MINTR32
C                                                                         023840   SP-MINTR33
      KTEST(16,KRF)=KTEST(16,KRF)+1                                       023850   SP-MINTR34
      IF (ITGTPS.NE.0) GO TO 130                                          023860   SP-MINTR35
C                                                                         023870   SP-MINTR36
      IABRT=16                                                            023880   SP-MINTR37
      CALL ABRTTL (IABRT)                                                 023890   SP-MINTR38
C                                                                         023900   SP-MINTR39
  130 RETURN                                                              023910   SP-MINTR40
      END                                                                 023920   SP-MINTR41

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 57 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
      ' NUTINE NOREC                                                          023940  SP-NOREC 2
C                                                                             023950  SP-NOREC 3
         'ON /ICODE/ ICODE                                                    000450  CM+ICODE 2
C                                                                             023970  SP-NOREC 5
      WRITE (6,100) ICODE                                                     023980  SP-NOREC 6
C                                                                             023990  SP-NOREC 7
      CALL STPACE                                                             024000  SP-NOREC 8
C                                                                             024010  SP-NOREC 9
      STOP ' I'M NOREC: ATTEMPTED TO READ NON-EXISTENT RECORD '               024020  SP-NOREC10
C                                                                             024030  SP-NOREC11
C                                                                             024040  SP-NOREC12
C *** F O R M A T   S T A T E M E N T S ***                                   024050  SP-NOREC13
C                                                                             024060  SP-NOREC14
  100 FORMAT (1H ,38H INPUT ATTEMPTED TO READ RECORD NAMED ,A10,30H BUT       024070  SP-NOREC15
     1 NO SUCH RECORD WAS FOUND)                                              024080  SP-NOREC16
      END                                                                     024090  SP-NOREC17
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 58 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE OUTPUT                                                 024110  SP-OUTPUT2
C                                                                       024120  SP-OUTPUT3
      COMMON /ABORT/ NABORT(20,6), NSABLM                               000110  CM+ABORT 2
      COMMON /ABRLBL/ IEABRT(3,19)                                      000130  CM+ABRLBL2
      COMMON /DODF/ DFDOKL(10,3), NDFSP, NDOKIL, DOKILD                 000250  CM+DODF  2
      LOGICAL DOKILD                                                    000260  CM+DODF  3
      COMMON /FLAG/ ENDREP, IDN, IPPFPL, IDIGIL, IPPTY, IPNDOC, IT3FLG, 000300  CM+FLAG  2
     1 ENDFLG                                                           000310  CM+FLAG  3
      LOGICAL ENDFLG, ENDREP                                            000320  CM+FLAG  4
      COMMON /KTEST/ KTEST(20,6)                                        000490  CM+KTEST 2
      COMMON /RPLCTN/ KREP, NREP, KRF, NRF, NVEHKL                      000770  CM+RPLCTN2
      COMMON /RUNDAT/ DAT, TIM, NCASE, ICCPFG, IPPFG                    000840  CM+RUNDAT2
      COMMON /SCCSS/ PNDPFC(6)                                          000900  CM+SCCSS 2
      DIMENSION NKILL(6), OUT(6)                                        024210  SP-OUTPU12
      DIMENSION ABRPCT(20,6), NTEST(20,6), PASPCT(20,6)                 024220  SP-OUTPU13
      DIMENSION PPRKO(6), PPRKAT(6), PPRKS(6)                           024230  SP-OUTPU14
      DIMENSION PRULOS(6)                                               024240  SP-OUTPU15
C                                                                       024250  SP-OUTPU16
      DO 110 J=1,6                                                      024260  SP-OUTPU17
      NKILL(J)=0                                                        024270  SP-OUTPU18
      OUT(J)=0.0                                                        024280  SP-OUTPU19
      DO 100 I=1,20                                                     024290  SP-OUTPU20
      NTEST(I,J)=0                                                      024300  SP-OUTPU21
  100 CONTINUE                                                          024310  SP-OUTPU22
  110 CONTINUE                                                          024320  SP-OUTPU23
C                                                                       024330  SP-OUTPU24
      DO 120 I=1,19                                                     024340  SP-OUTPU25
      DO 120 J=1,6                                                      024350  SP-OUTPU26
      ABRPCT(I,J)=-1.0                                                  024360  SP-OUTPU27
      IF (NREP.NE.0) ABRPCT(I,J)=100.*FLOAT(NABORT(I,J))/FLOAT(NREP)    024370  SP-OUTPU28
  120 CONTINUE                                                          024380  SP-OUTPU29
C                                                                       024390  SP-OUTPU30
      DO 150 J=1,6                                                      024400  SP-OUTPU31
      NTEST(1,J)=NREP                                                   024410  SP-OUTPU32
      DO 130 I=2,19                                                     024420  SP-OUTPU33
      NTEST(I,J)=NTEST(I-1,J)-NABORT(I-1,J)                             024430  SP-OUTPU34
  130 CONTINUE                                                          024440  SP-OUTPU35
      DO 140 I=1,19                                                     024450  SP-OUTPU36
      IF (NTEST(I,J).EQ.0) PASPCT(I,J)=-1.0                             024460  SP-OUTPU37
      IF (NTEST(I,J).NE.0) PASPCT(I,J)=100.*(1.-FLOAT(NABORT(I,J))      024470  SP-OUTPU38
     1 /FLOAT(NTEST(I,J)))                                              024480  SP-OUTPU39
  140 CONTINUE                                                          024490  SP-OUTPU40
  150 CONTINUE                                                          024500  SP-OUTPU41
C                                                                       024510  SP-OUTPU42
      IDCSN=NREP                                                        024520  SP-OUTPU43
      IATFNG=NTEST(8,1)                                                 024530  SP-OUTPU44
      ISHOT=NTEST(12,1)                                                 024540  SP-OUTPU45
C                                                                       024550  SP-OUTPU46
      DO 160 J=1,6                                                      024560  SP-OUTPU47
      NKILL(J)=NTEST(19,J)-NABORT(19,J)                                 024570  SP-OUTPU48
      IF (PNDPFC(J).NE.FLOAT(NKILL(J))) WRITE (6,300) J,PNDPFC(J),NKILL(024580  SP-OUTPU49
     1J)                                                                024590  SP-OUTPU50
  160 CONTINUE                                                          024600  SP-OUTPU51
C                                                                       024610  SP-OUTPU52

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 59 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
      DO 170 J=1,6                                                      024620  SR-OUTPU53
      PPBKO(J)=-1.0                                                     024630  SP-OUTPU54
      PPBKAE(J)=-1.0                                                    024640  SP-OUTPU55
      PPBKS(J)=-1.0                                                     024650  SR-OUTPU56
      IF (IOCSN.NE.0) PPBKO(J)=100.*FLOAT(NKILL(J))/FLOAT(IOCSN)        024660  SR-OUTPU57
      IF (IATENG.NE.0) PPBKAE(J)=100.*FLOAT(NKILL(J))/FLOAT(IATENG)     024670  SP-OUTPU58
      IF (ISHOT.NE.0) PPBKS(J)=100.*FLOAT(NKILL(J))/FLOAT(ISHOT)        024680  SP-OUTPU59
  170 CONTINUE                                                          024690  SP-OUTPU60
C                                                                       024700  SP-OUTPU61
      PRBSO=-1.0                                                        024710  SP-OUTPU62
      PPBSAE=-1.0                                                       024720  SP-OUTPU63
      PRBAFO=-1.0                                                       024730  SP-OUTPU64
      IF (IOCSN.NE.0) PRBSO=100.*FLOAT(ISHOT)/FLOAT(IOCSN)              024740  SP-OUTPU65
      IF (IATENG.NE.0) PPBSAE=100.*FLOAT(ISHOT)/FLOAT(IATENG)           024750  SP-OUTPU66
      IF (IOCSN.NE.0) PRBAFO=100.*FLOAT(IATENG)/FLOAT(IOCSN)            024760  SP-OUTPU67
C                                                                       024770  SP-OUTPU68
      DO 180 J=1,6                                                      024780  SP-OUTPU69
      PPNLOS(J)=PASPCT(11,J)*PASPCT(12,J)/100.                          024790  SR-OUTPU70
  180 CONTINUE                                                          024800  SR-OUTPU71
C                                                                       024810  SP-OUTPU72
      DO 190 II=1,20                                                    024820  SP-OUTPU73
      LIMU=1                                                            024830  SP-OUTPU74
      IF (II.GT.MSABLM) LIMU=6                                          024840  SP-OUTPU75
      DO 190 JJ=1,LIMU                                                  024850  SP-OUTPU76
      IF (NTEST(II,JJ).EQ.KTEST(II,JJ)) GO TO 190                       024860  SP-OUTPU77
      IF (IFNDOC.EQ.1.AND.(II.EQ.4.OR.II.EQ.11).AND.KTEST(II,JJ).EQ.0) G024870  SP-OUTPU78
     1O TO 190                                                          024880  SP-OUTPU79
      WRITE (6,290) II,JJ                                               024890  SP-OUTPU80
      WRITE (6,310) ((KTEST(II,JL),JL=1,6),II=1,20)                     024900  SR-OUTPU81
      GO TO 200                                                         024910  SP-OUTPU82
  190 CONTINUE                                                          024920  SP-OUTPU83
  200 CONTINUE                                                          024930  SP-OUTPU84
C                                                                       024940  SP-OUTPU85
      PPOOKL=-1.0                                                       024950  SP-OUTPU86
      PPDOKC=-1.0                                                       024960  SP-OUTPU87
      IF (NREP.NE.0) PRDOKL=100.*FLOAT(NDOKIL)/FLOAT(NREP)              024970  SP-OUTPU88
      IF (NTEST(14,1).NE.0) PPDOKC=100.*(1.-FLOAT(NDOKIL)               024980  SP-OUTPU89
     1  /FLOAT(NTEST(14,1)))                                            024990  SP-OUTPU90
C                                                                       025000  SP-OUTPU91
      WRITE (6,320)                                                     025010  SP-OUTPU92
      WRITE (6,330) IOCSN                                               025020  SP-OUTPU93
C                                                                       025030  SP-OUTPU94
      DO 210 I=1,11                                                     025040  SP-OUTPU95
      X=ABFPCT(I,1)                                                     025050  SR-OUTPU96
      XOUT=FPRCNT(X,.TRUE.)                                             025060  SR-OUTPU97
      Y=PASPCT(I,1)                                                     025070  SP-OUTPU98
      YOUT=FPRCNT(Y,.TRUE.)                                             025080  SP-OUTPU99
      WRITE (6,340) I,(IDARPT(K,I),K=1,3),NARDPT(I,1),XOUT,NTEST(I,1),  025090  SP-OUTP100
     1  YOUT                                                            025100  SP-OUTP101
      IF (I.EQ.7) WRITE (6,390) IATENG                                  025110  SP-OUTP102
      IF (I.EQ.11) WRITE (6,400) ISHOT                                  025120  SP-OUTP103
  210 CONTINUE                                                          025130  SP-OUTP104
C                                                                       025140  SP-OUTP105
      WRITE (6,350)                                                     025150  SP-OUTP106
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 60 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
C                                                                        025160   SP-OUTP107
      DO 240 I=12,19                                                     025170   SP-OUTP108
      DO 220 J=1,NRF                                                     025180   SP-OUTP109
      X=ABRPCT(I,J)                                                      025190   SP-OUTP110
      OUT(J)=FPRCNT(X,.TRUE.)                                            025200   SP-OUTP111
  220 CONTINUE                                                           025210   SP-OUTP112
      WRITE (6,360) I,(IDABPT(K,I),K=1,3),(MABORT(I,J),OUT(J),J=1,NPF)   025220   SP-OUTP113
      DO 230 J=1,NRF                                                     025230   SP-OUTP114
      X=PASPCT(I,J)                                                      025240   SP-OUTP115
      OUT(J)=FPRCNT(X,.TRUE.)                                            025250   SP-OUTP116
  230 CONTINUE                                                           025260   SP-OUTP117
      WRITE (6,370) (NTEST(I,J),OUT(J),J=1,NPF)                          025270   SP-OUTP118
  240 CONTINUE                                                           025280   SP-OUTP119
C                                                                        025290   SP-OUTP120
      WRITE (6,380) (NKILL(J),J=1,NPF)                                   025300   SP-OUTP121
C                                                                        025310   SP-OUTP122
      DO 250 J=1,NRF                                                     025320   SP-OUTP123
      X=PRBKD(J)                                                         025330   SP-OUTP124
      OUT(J)=FPRCNT(X,.FALSE.)                                           025340   SP-OUTP125
  250 CONTINUE                                                           025350   SP-OUTP126
      WRITE (6,410) (OUT(I),I=1,NPF)                                     025360   SP-OUTP127
C                                                                        025370   SP-OUTP128
      DO 260 J=1,NPF                                                     025380   SP-OUTP129
      X=PRBKAE(J)                                                        025390   SP-OUTP130
      OUT(J)=FPRCNT(X,.FALSE.)                                           025400   SP-OUTP131
  260 CONTINUE                                                           025410   SP-OUTP132
      WRITE (6,420) (OUT(J),J=1,NPF)                                     025420   SP-OUTP133
C                                                                        025430   SP-OUTP134
      DO 270 J=1,NRF                                                     025440   SP-OUTP135
      X=PPBKS(J)                                                         025450   SP-OUTP136
      OUT(J)=FPRCNT(X,.FALSE.)                                           025460   SP-OUTP137
  270 CONTINUE                                                           025470   SP-OUTP138
      WRITE (6,430) (OUT(J),J=1,NPF)                                     025480   SP-OUTP139
C                                                                        025490   SP-OUTP140
      DO 280 J=1,NRF                                                     025500   SP-OUTP141
      X=PPHLOS(J)                                                        025510   SP-OUTP142
      OUT(J)=FPRCNT(X,.FALSE.)                                           025520   SP-OUTP143
  280 CONTINUE                                                           025530   SP-OUTP144
      WRITE (6,450) (OUT(J),J=1,NPF)                                     025540   SP-OUTP145
C                                                                        025550   SP-OUTP146
      XOUT=FPRCNT(PRBSO,.FALSE.)                                         025560   SP-OUTP147
      YOUT=FPRCNT(PPBSAE,.FALSE.)                                        025570   SP-OUTP148
      ZOUT=FPRCNT(PPBAFO,.FALSE.)                                        025580   SP-OUTP149
      WRITE (6,440) XOUT,YOUT,ZOUT                                       025590   SP-OUTP150
C                                                                        025600   SP-OUTP151
      XOUT=FPRCNT(PRDOKL,.TRUE.)                                         025610   SP-OUTP152
      YOUT=FPRCNT(PRDOKC,.TRUE.)                                         025620   SP-OUTP153
      WRITE (6,460) NOOKIL,XOUT,NTEST(14,1),YOUT                         025630   SP-OUTP154
C                                                                        025640   SP-OUTP155
      RETURN                                                             025650   SP-OUTP156
C                                                                        025660   SP-OUTP157
C                                                                        025670   SP-OUTP158
C  * * *   F O R M A T   S T A T E M E N T S   * * *                     025680   SP-OUTP159
C                                                                        025690   SP-OUTP160

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 61 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

  290 FORMAT (56H SERIOUS PROBLEM !  KTEST AND NTEST ARRAYS DO NOT AGREE025700  SR-OUTP161
     1.,4H II=,I5,4H JJ=,I5)                                             025710  SR-OUTP162
  300 FORMAT (1H ,24H IN OUTPUT, PROBLEM 1 J=,I5,9H  ENDPEC=,F5.0,8H  NK025720  SR-OUTP163
     1ILL=,I5)                                                           025730  SP-OUTP164
  310 FORMAT (1H ,12HKTEST ARRAY:,/,(1H ,6I10))                          025740  SP-OUTP165
  320 FORMAT (1H1,14X,14HCAUSE OF ABORT)                                 025750  SP-OUTP166
  330 FORMAT (50X,5H*****,1X,10HOCCASIONS ,1H(,I5,1H),6H *****)         025760  SR-OUTP167
  340 FORMAT (1H ,/,1H ,I2,3X,3A10,I7,1H(,A7,5X,25HNUMBER TESTED (% PASS025770  SR-OUTP168
     1ED):,I7,1H(,A7)                                                    025780  SP-OUTP169
  350 FORMAT (1H ,7X,14HROUND NUMBER :,23X,1H1,14X,1H2,14X,1H3,14X,1H4,1025790  SP-OUTP170
     14X,1H5,14X,1H6)                                                    025800  SP-OUTP171
  360 FORMAT (1H ,/,1H ,I2,3X,3A10,6(I7,1H(,A7))                         025810  SP-OUTP172
  370 FORMAT (8X,24HNUMBER TESTED (% PASSED),4X,6(I7,1H(,A7))            025820  SP-OUTP173
  380 FORMAT (1H ,/,15X,11H***** KILLS,5X,6(10X,I5),6H *****)            025830  SP-OUTP174
  390 FORMAT (1H ,/,45X,5H*****,1X,23HATTEMPTED ENGAGEMENTS (,I5,7H) ***025840  SR-OUTP175
     1**)                                                                025850  SP-OUTP176
  400 FORMAT (1H ,/,53X,6H***** ,7HSHOTS (,I5,7H) *****)                 025860  SR-OUTP177
  410 FORMAT (1H ,/,8X,30HPROB(KILL/DCSN)=                   ,1X,6(A10,5X))  025870  SP-OUTP178
  420 FORMAT (8X,30HPROB(KILL/ATMP ENG)=          ,1X,6(A10,5X))         025880  SP-OUTP179
  430 FORMAT (8X,30HPROB(KILL/SHOT)=              ,1X,6(A10,5X))         025890  SP-OUTP180
  440 FORMAT (8X,16HPROB(SHOT/DCSN)=,1X,A10,8X,20HPROB(SHOT/ATMP ENG)=,1025900  SP-OUTP181
     1X,A10,8X,20HPROB(ATMP ENG/DCSN)=,1X,A10)                           025910  SP-OUTP182
  450 FORMAT (8X,30HPROB(LOS)=                    ,1X,6(A10,5X))         025920  SP-OUTP183
  460 FORMAT (1H ,/,1H ,61HDESIGNATOR SURVIVABILITY: CASES IN WHICH DESI025930  SP-OUTP184
     1GNATOR KILLED = ,I5,1H(,A7,5X,30HNUMBER TESTED (% NOT KILLED) =,I5025940  SP-OUTP185
     2,1H(,A7)                                                           025950  SP-OUTP186
      END                                                                025960  SP-OUTP187

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1  FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 62 Of 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE PECHK                                                  025980   SP-PECHK  2
C                                                                       025990   SP-PECHK  3
      COMMON /FLAG/ ENDREP, IDN, IPPEPL, IDIGTL, IPPTY, IPNDUC, IT3FLG, 000300   CM+FLAG   2
     1  ENDFLG                                                          000310   CM+FLAG   3
      LOGICAL ENDFLG, ENDREP                                            000320   CM+FLAG   4
      COMMON /HIT/ INDEX(6,5,2), PETBL(60,10,7), DLTT(6), PKTBL(10,2),  000410   CM+HIT    2
     1  NRNGTT, RNGTTF(20), TTF(3,20,2), NPNGPS, PNGPST(10), PSTTPL(10, 000420   CM+HIT    3
     2  3)                                                              000430   CM+HIT    4
      COMMON /KTEST/ KTEST(20,6)                                        000490   CM+KTEST  2
      COMMON /RANDOM/ IR1, IP2, IP3, IP4, IP5, IP6, IP7, IP8, IP9, IP10,000640   CM+RANDOM2
     1  IP11, IR12, IP13, IP14, IR15, IP16, IP17, IR18, IP19, IP20, IP21000650   CM+RANDOM3
     2  , IP22, IP23, IP24, IP25, IP26, IP27, IP28                      000660   CM+RANDOM4
      COMMON /RANGE/ DTRNG, MXDPNG, VISLIM, PNGNOW                      000680   CM+RANGE  2
      REAL MXDRNG                                                       000690   CM+RANGE  3
      COMMON /RPLCTN/ KREP, NREP, KPF, NRF, NVEHKL                      000770   CM+RPLCTN2
      COMMON /RSPTIM/ RATRTM(3), XMVADF(2), DUM1, RCSPTM(2,2), XMTTIM(2,000790   CM+RSPTIM2
     1  2), TMEAN(2,2), TSIGMA(2,2), DETTMA(10,2), NDT, TRAPPY(3,2),    000800   CM+RSPTIM3
     2  PRSPTM, TRBAR, PPSDT, PESVT, ALPHA(2,2), BETA(2,2), DETTIM,      000810   CM+PSPTIM4
     3  IDSGTP, RUMC, PCRMSG, PRDOW                                     000820   CM+RSPTIM5
      COMMON /TARGET/ TVEL, NVEHCL, DISRVH, ITGTTP, IMUTS, ITGTPS       001000   CM+TARGET2
      COMMON /TIME/ DUPLOS, TIMNOW, TOF, TBP, TIMPA, TIMPA1             001020   CM+TIME   2
      COMMON /WEATHR/ PRCLCG, PPCFLS(2), PPGCFL(11,2), W(6,11,2),       001060   CM+WEATHR2
     1  VIS(11), CLOUD(6), NCC, NVL, PASQL(6), WNDSPD(3,3), HUMID(2),   001070   CM+WEATHR3
     2  WSUM(2), WCFSUM(2), ICC, IVL                                    001080   CM+WEATHR4
C                                                                       026100   SP-PECHK14
      RNGK=RNGNOW/1000.                                                 026110   SP-PECHK15
C                                                                       026120   SR-PECHK16
      T1=FLOAT(KRF-1)*TBR-FLOAT(NVEHCL-1)*DISRVH/TVEL                   026130   SR-PECHK17
      T2=TIMPA1-DETTIM-TRBAR                                            026140   SR-PECHK18
      T3=0.0                                                            026150   SR-PECHK19
      IF (IPREPL.EQ.1.AND.IT3FLG.EQ.1) T3=TRBAR+DETTIM-RUMC/TVEL        026160   SP-PECHK20
      DELTAT=T1+T2+T3                                                   026170   SP-PECHK21
      IF (DELTAT.LE.0.0) DELTAT=0.0                                     026180   SR-PECHK22
C                                                                       026190   SR-PECHK23
      IMUTS=1                                                           026200   SP-PECHK24
      IF (DELTAT.LE.0.0) IMUTS=2                                        026210   SP-PECHK25
C                                                                       026220   SR-PECHK26
C  FIND IRNG1 AND IRNG2 WHICH BRACKET RNGK.                             026230   SP-PECHK27
C  IF RNGK<1, IRNG1=IRNG2=1; IF RNGK>7, IRNG1=IRNG2=7.                  026240   SP-PECHK28
C  OTHERWISE, IRNG1 <= RNGK < IRNG2.                                    026250   SR-PECHK29
C  IRNG1 AND IRNG2 WILL BE USED IN INTERPOLATING IN PETBL.              026260   SP-PECHK30
C                                                                       026270   SP-PECHK31
      IRNG1=MIN1(RNGK,7.0)                                              026280   SP-PECHK32
      IRNG1=MAX0(1,IRNG1)                                               026290   SP-PECHK33
      IRNG2=IRNG1+1                                                     026300   SP-PECHK34
      IF (RNGK.LT.1.0.OR.IRNG1.EQ.7) IRNG2=IRNG1                        026310   SP-PECHK35
      TF=AMAX1(RNGK-FLOAT(IRNG1),0.0)                                   026320   SP-PECHK36
      IF (TF.EQ.0.0) GO TO 100                                          026330   SP-PECHK37
      TF=TF/(FLOAT(IRNG2)-FLOAT(IRNG1))                                 026340   SP-PECHK38
  100 IF (IMUTS.NE.2) GO TO 110                                         026350   SP-PECHK39
C                                                                       026360   SP-PECHK40
      IDELTT=1                                                          026370   SP-PECHK41
      II=INDEX(ICC,IDELTT,IMUTS)                                        026380   SP-PECHK42
      IVLA=MINO(IVL,10)                                                 026390   SP-PECHK43

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 63 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789O (READ VERTICALLY)

      PE1=PETBL(II,IVLA,IPNG1)                                          026400  SP-PECHK44
      PE2=PETBL(II,IVLA,IPNG2)                                          026410  SP-PECHK45
      PE=PE1+TF*(PE2-PE1)                                               026420  SP-PECHK46
      GO TO 140                                                         026430  SP-PECHK47
C                                                                       026440  SP-PECHK48
C  BRACKET DELTAT WITH DLTT VALUES CORRESPONDING TO IDELT1 AND IDELT2   026450  SP-PECHK49
C                                                                       026460  SP-PECHK50
  110 DO 120 I=1,NDT                                                    026470  SP-PECHK51
      IF (DELTAT.LT.DLTT(I).OR.DELTAT.GE.DLTT(I+1)) GO TO 120           026480  SP-PECHK52
      IDELT1=I                                                          026490  SP-PECHK53
      IDELT2=I+1                                                        026500  SP-PECHK54
      GO TO 130                                                         026510  SP-PECHK55
  120 CONTINUE                                                          026520  SP-PECHK56
      STOP ' IN PECHK: FAILED TO BRACKET DELTA T '                      026530  SP-PECHK57
C                                                                       026540  SP-PECHK58
C  PERFORM A TWO DIMENSIONAL FOUR-POINT INTERPOLATION USING TWO RANGE   026550  SP-PECHK59
C  POINTS FOR EACH OF TWO DELAY TIME VALUES.                            026560  SP-PECHK60
C                                                                       026570  SP-PECHK61
  130 FDT=DELTAT-DLTT(IDELT1)                                           026580  SP-PECHK62
      FDT2=FDT/(DLTT(IDELT2)-DLTT(IDELT1))                              026590  SP-PECHK63
      FDT1=1.0-FDT2                                                     026600  SP-PECHK64
      TF2=TF                                                            026610  SP-PECHK65
      TF1=1.0-TF2                                                       026620  SP-PECHK66
      II1=INDEX(ICC,IDELT1,IMUTS)                                       026630  SP-PECHK67
      II2=INDEX(ICC,IDELT2,IMUTS)                                       026640  SP-PECHK68
      IVLA=MIN0(10,IVL)                                                 026650  SP-PECHK69
      PET1P1=PETBL(II1,IVLA,IPNG1)                                      026660  SP-PECHK70
      PET1P2=PETBL(II1,IVLA,IPNG2)                                      026670  SP-PECHK71
      PET2R1=PETBL(II2,IVLA,IPNG1)                                      026680  SP-PECHK72
      PET2P2=PETBL(II2,IVLA,IPNG2)                                      026690  SP-PECHK73
      IF (IDELT2.EQ.6) PET2P1=0.0                                       026700  SP-PECHK74
      IF (IDELT2.EQ.6) PET2P2=0.0                                       026710  SP-PECHK75
      PE=FDT1*TF1*PET1P1+FDT1*TF2*PET1P2+FDT2*TF1*PET2R1+FDT2*TF2*PET2R2026720  SP-PECHK76
C                                                                       026730  SP-PECHK77
  140 RN22=URAN31(IR22)                                                 026740  SP-PECHK78
      KTEST(17,KRF)=KTEST(17,KRF)+1                                     026750  SP-PECHK79
      IF (RN22.LE.PE) GO TO 150                                         026760  SP-PECHK80
C                                                                       026770  SP-PECHK81
      IABRT=17                                                          026780  SP-PECHK82
      CALL ABRTTL (IABRT)                                               026790  SP-PECHK83
C                                                                       026800  SP-PECHK84
  150 RETURN                                                            026810  SP-PECHK85
      END                                                               026820  SP-PECHK86

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789O (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 64 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE PENAME (TP,IDT,VEL,GTP,REFL,ANGLET,DEFB,TGTHD,SKSEN,          026840   SR-PENAME2
     1  IDCODE)                                                                 026850   SP-PENAME3
C                                                                               026860   SR-PENAME4
      COMMON /XVALUE/ XVALUE(8,9,2)                                             001100   CM*XVALUE2
      DIMENSION AVALUE(9), NP(9)                                                026880   SP-PENAME6
C                                                                               026890   SP-PENAME7
C                                                                               026900   SP-PENAME8
      AVALUE(1)=TP                                                              026910   SP-PENAME9
      AVALUE(2)=IDT                                                             026920   SP-PENAM10
      AVALUE(3)=VEL                                                             026930   SR-PENAM11
      AVALUE(4)=GTP                                                             026940   SR-PENAM12
      AVALUE(5)=REFL                                                            026950   SP-PENAM13
      AVALUE(6)=ANGLET                                                          026960   SP-PENAM14
      AVALUE(7)=DEFB                                                            026970   SP-PENAM15
      AVALUE(8)=TGTHD                                                           026980   SP-PENAM16
      AVALUE(9)=SKSEN                                                           026990   SR-PENAM17
C                                                                               027000   SP-PENAM18
      DO 110 J=2,9                                                              027010   SP-PENAM19
      DO 100 I=1,8                                                              027020   SP-PENAM20
      IF (ABS(AVALUE(J)-XVALUE(I,J,1)).GT..0001) GO TO 100                      027030   SP-PENAM21
      NP(J)=XVALUE(I,J,2)+.5                                                    027040   SP-PENAM22
      GO TO 110                                                                 027050   SR-PENAM23
  100 CONTINUE                                                                  027060   SP-PENAM24
      STOP ' IN PENAME: ERROR NUMBER 1 '                                        027070   SP-PENAM25
  110 CONTINUE                                                                  027080   SP-PENAM26
C                                                                               027090   SR-PENAM27
      DO 120 J=2,9                                                              027100   SP-PENAM28
      IF (NP(J).GE.8) STOP ' IN PENAME: ERROR NUMBER 2 '                        027110   SR-PENAM29
  120 CONTINUE                                                                  027120   SP-PENAM30
C                                                                               027130   SP-PENAM31
      IF (TP.LE.0.0.OR.TP.GE.999.5) STOP ' IN PENAME: ERROR NUMBER 3 '          027140   SP-PENAM32
      NCODE=TP+.50                                                              027150   SR-PENAM33
C                                                                               027160   SP-PENAM34
      DO 130 J=2,9                                                              027170   SP-PENAM35
  130 NCODE=8*NCODE+NP(J)                                                       027180   SR-PENAM36
C                                                                               027190   SP-PENAM37
      ENCODE (10,170,IDCODE) NCODE                                              027200   SP-PENAM38
C                                                                               027210   SR-PENAM39
      RETURN                                                                    027220   SP-PENAM40
C                                                                               027230   SR-PENAM41
C  *    *    *    *    *    *    *    *    *    *    *    *                     027240   SP-PENAM42
C                                                                               027250   SP-PENAM43
      ENTRY PEIDNT                                                              027260   SP-PENAM44
C                                                                               027270   SR-PENAM45
      DECODE (10,180,IDCODE) NCODE                                              027280   SP-PENAM46
C                                                                               027290   SP-PENAM47
      DO 140 J=2,9                                                              027300   SP-PENAM48
      NP(11-J)=MOD(NCODE,8)                                                     027310   SR-PENAM49
      NCODE=NCODE/8                                                             027320   SP-PENAM50
  140 CONTINUE                                                                  027330   SP-PENAM51
C                                                                               027340   SP-PENAM52
      NP(1)=NCODE                                                               027350   SP-PENAM53
C                                                                               027360   SP-PENAM54
      DO 160 J=2,9                                                              027370   SP-PENAM55

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 65 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      DO 150 I=1,8                                                     027380   SP-PENAM56
      IF (ABS(FLOAT(NP(J))-XVALUE(I,J,2)).GT..0001) GO TO 150           027390   SP-PENAM57
      AVALUE(J)=XVALUE(I,J,1)                                          027400   SP-PENAM58
      GO TO 160                                                        027410   SP-PENAM59
  150 CONTINUE                                                         027420   SP-PENAM60
      STOP ' IN PENAME (PEIDNT): ERROR NUMBER 4 '                      027430   SP-PENAM61
  160 CONTINUE                                                         027440   SP-PENAM62
C                                                                      027450   SP-PENAM63
      AVALUE(1)=NCODE                                                  027460   SP-PENAM64
C                                                                      027470   SP-PENAM65
      TR=AVALUE(1)                                                     027480   SP-PENAM66
      IDT=AVALUE(2)                                                    027490   SP-PENAM67
      VEL=AVALUE(3)                                                    027500   SP-PENAM68
      GTP=AVALUE(4)                                                    027510   SP-PENAM69
      REFL=AVALUE(5)                                                   027520   SP-PENAM70
      ANGLFT=AVALUE(6)                                                 027530   SP-PENAM71
      DEFB=AVALUE(7)                                                   027540   SP-PENAM72
      TGTHD=AVALUE(8)                                                  027550   SP-PENAM73
      SKSFN=AVALUE(9)                                                  027560   SP-PENAM74
C                                                                      027570   SP-PENAM75
      RETURN                                                           027580   SP-PENAM76
C                                                                      027590   SP-PENAM77
C                                                                      027600   SP-PENAM78
C   * * *   F O R M A T    S T A T E M E N T S   * * *                 027610   SP-PENAM79
C                                                                      027620   SP-PENAM80
  170 FORMAT (4HPE00,P6)                                               027630   SP-PENAM81
  180 FORMAT (4X,P6)                                                   027640   SP-PENAM82
      END                                                              027650   SP-PENAM83

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 66 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      SUBROUTINE PKCHK                                                  027670  SP-PKCHK 2
C                                                                       027680  SP-PKCHK 3
      COMMON /HIT/ INDEX(6,5,2), PETBL(60,10,7), DLTT(6), PKTBL(10,2),  000410  CM+HIT   2
     1  NRNGTT, RNGTTF(20), TTF(3,20,2), NRNGPS, RNGPST(10), PSTTBL(10, 000420  CM+HIT   3
     2  3)                                                              000430  CM+HIT   4
      COMMON /KTEST/ KTEST(20,6)                                        000490  CM+KTEST 2
      COMMON /RANDOM/ IR1, IR2, IR3, IR4, IR5, IR6, IR7, IR8, IR9, IR10,000640  CM+RANDOM2
     1  IR11, IR12, IR13, IR14, IR15, IR16, IR17, IR18, IR19, IR20, IR21000650  CM+RANDOM3
     2  , IR22, IR23, IR24, IR25, IR26, IR27, IR28                      000660  CM+RANDOM4
      COMMON /RPLCTN/ KREP, NREP, KRF, NPF, NVEHKL                      000770  CM+RPLCTN2
      COMMON /SCCSS/ RNDREC(6)                                          000900  CM+SCCSS 2
      COMMON /TARGET/ TVEL, NVEHCL, DISRVH, ITGTTP, IMUTS, ITGTPS       001000  CM+TARGET2
C                                                                       027750  SR-PKCHK10
      PK=PKTBL(ITGTTP,ITGTPS)                                           027760  SR-PKCHK11
      RN21=URAN31(IR21)                                                 027770  SR-PKCHK12
      KTEST(19,KRF)=KTEST(19,KPF)+1                                     027780  SR-PKCHK13
      IF (RN21.LE.PK) GO TO 100                                         027790  SR-PKCHK14
C                                                                       027800  SR-PKCHK15
      IABRT=19                                                          027810  SP-PKCHK16
      CALL ABRTTL (IABRT)                                               027820  SR-PKCHK17
      GO TO 110                                                         027830  SR-PKCHK18
C                                                                       027840  SP-PKCHK19
  100 RNDREC(KRF)=RNDREC(KPF)+1.0                                       027850  SR-PKCHK20
      NVEHKL=MINO(NVEHKL+1,NVEHCL)                                      027860  SR-PKCHK21
C                                                                       027870  SR-PKCHK22
  110 RETURN                                                            027880  SR-PKCHK23
C                                                                       027890  SR-PKCHK24
      END                                                               027900  SP-PKCHK25

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 67 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE PPLOT                                                  027920  SR-PPLOT 2
C                                                                       027930  SR-PPLOT 3
      COMMON /PUNDAT/ DAT, TIM, NCASE, ICCPEG, IPPEG                    000840  CM+PUNDAT2
C                                                                       027950  SR-PPLOT 5
      DIMENSION CM(10)                                                  027960  SR-PPLOT 6
C                                                                       027970  SR-PPLOT 7
      ENCODE (100,100,CM(1) )NCASE,IPPEG                                027980  SR-PPLOT 8
      CALL COMMNT (CM,10,.FALSE.,.FALSE.)                               027990  SR-PPLOT 9
      RETURN                                                            028000  SR-PPLOT10
C                                                                       028010  SR-PPLOT11
C                                                                       028020  SR-PPLOT12
C  * * *  F O R M A T   S T A T E M E N T S  * * *                      028030  SR-PPLOT13
C                                                                       028040  SR-PPLOT14
  100 FORMAT (1H ,19H IN PPLOT FOR CASE ,I2,13H WITH IPPEG= ,I1,23H , N028050  SR-PPLOT15
     1O PLOT PRODUCED . )                                               028060  SR-PPLOT16
      END                                                               028070  SR-PPLOT17

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 68 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE PSMOKE                                                  028090  SP-PSMOKE2
C                                                                        028100  SP-PSMOKE3
C                                                                        028110  SP-PSMOKE4
      COMMON /SMOKED/ SMK2(3,3,2), SMK5(3,3,2), FRONT, NSMK2, NSMK5,     000920  CM+SMOKED2
     1  PSMKKL                                                           000930  CM+SMOKED3
      COMMON /WEATHR/ PPCLCG, PPCFLS(2), PPGCFL(11,2), W(6,11,2),        001060  CM+WEATHR2
     1  VIS(11), CLOUD(6), ICC, IVI, PASQL(6), WNDSPD(3,3), HUMID(2),    001070  CM+WEATHR3
     2  WSUM(2), WCFSUM(2), ICC, IVI                                     001080  CM+WEATHR4
      PSMKKL=0.0                                                         028140  SP-PSMOKE7
C                                                                        028150  SP-PSMOKE8
      DO 100 I=1,3                                                       028160  SP-PSMOKE9
      JEND=I                                                             028170  SP-PSMOK10
      IF (I.EQ.2) JEND=3                                                 028180  SP-PSMOK11
      DO 100 J=1,JEND                                                    028190  SP-PSMOK12
      DO 100 K=1,2                                                       028200  SP-PSMOK13
      VSMK=FLOAT(NSMK2)/SMK2(I,J,K)+FLOAT(NSMK5)/SMK5(I,J,K)             028210  SP-PSMOK14
      VSMK=AMIN1(FRONT,VSMK)                                             028220  SR-PSMOK15
      PASQT=PASQL(2*I-1)+PASQL(2*I)                                      028230  SP-PSMOK16
      PSMKKL=PSMKKL+PASQT*WNDSPD(I,J)*HUMID(K)*VSMK                      028240  SR-PSMOK17
  100 CONTINUE                                                           028250  SP-PSMOK18
C                                                                        028260  SP-PSMOK19
      PSMKKL=PSMKKL/FRONT                                                028270  SP-PSMOK20
C                                                                        028280  SP-PSMOK21
      RETURN                                                             028290  SP-PSMOK22
      END                                                                028300  SP-PSMOK23

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 69 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      SUBROUTINE REINTZ                                                 028320   SP-REINTZ2
C                                                                       028330   SR-REINTZ3
      COMMON /ABORT/ NABORT(20,6), MSABLM                               000110   CM+ABORT 2
      COMMON /KTEST/ KTEST(20,6)                                        000490   CM+KTEST 2
      COMMON /RANDOM/ IR1, IP2, IP3, IP4, IP5, IR6, IR7, IR8, IP9, IP10,000640   CM+RANDOM2
     1  IP11, IR12, IP13, IP14, IP15, IP16, IP17, IP18, IP19, IP20, IR21000650   CM+RANDOM3
     2  , IR22, IR23, IP24, IP25, IP26, IP27, IP28                      000660   CM+RANDOM4
      COMMON /RPLCTN/ KFEP, NPFP, KPF, NRF, NVEHKL                      000770   CM+RPLCTN2
      COMMON /SCCSS/ PNERFC(6)                                          000900   CM+SCCSS 2
C                                                                       028390   SR-REINTZ9
      DIMENSION IR(28)                                                  028400   SP-REINT10
C                                                                       028410   SP-REINT11
      EQUIVALENCE (IR(1),IP1)                                           028420   SR-REINT12
C                                                                       028430   SP-REINT13
      DO 100 IKTEST=1,20                                                028440   SP-REINT14
      DO 100 JKTEST=1,6                                                 028450   SP-REINT15
  100 KTEST(IKTEST,JKTEST)=0                                            028460   SR-REINT16
      DO 120 J=1,NRF                                                    028470   SR-REINT17
      RNDRFC(J)=0.0                                                     028480   SR-REINT18
      DO 110 I=1,20                                                     028490   SP-REINT19
      NABORT(I,J)=0                                                     028500   SR-REINT20
  110 CONTINUE                                                          028510   SP-REINT21
  120 CONTINUE                                                          028520   SP-REINT22
C                                                                       028530   SR-REINT23
      K=1                                                               028540   SP-REINT24
      M=2**25                                                           028550   SP-REINT25
      DO 130 I=1,28                                                     028560   SR-REINT26
      K=MOD(8*K,29)                                                     028570   SP-REINT27
      IR(I)=2*(M*K/29)+1                                                028580   SP-REINT28
  130 CONTINUE                                                          028590   SP-REINT29
C                                                                       028600   SP-REINT30
      IF (IPPINT.EQ.0) WRITE (6,140) (IP(I),I=1,28)                     028610   SP-REINT31
      IPRINT=1                                                          028620   SP-REINT32
      RETURN                                                            028630   SP-REINT33
C                                                                       028640   SP-REINT34
C                                                                       028650   SP-REINT35
C   * * *  F O R M A T   S T A T E M E N T S  * * *                     028660   SP-REINT36
C                                                                       028670   SP-REINT37
  140 FORMAT (1H ,21HRANDOM NUMBER SEEDS :,/,(10I10))                   028680   SP-REINT38
      END                                                               028690   SP-REINT39

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 70 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE RNDREL                                                 028710   SP-RNDREL2
C                                                                       028720   SP-RNDREL3
      COMMON /KTEST/ KTEST(20,6)                                        000490   CM+KTEST 2
      COMMON /MISC/ DDAPPK, PDSTKL, RELIF                               000540   CM+MISC  2
      COMMON /RANDOM/ IP1, IP2, IP3, IP4, IP5, IP6, IP7, IP8, IP9, IR10,000640   CM+RANDOM2
     1  IP11, IP12, IP13, IP14, IP15, IP16, IP17, IP18, IP19, IP20, IP21000650   CM+RANDOM3
     2  , IP22, IP23, IP24, IP25, IP26, IP27, IP28                      000660   CM+RANDOM4
      COMMON /FPLCTN/ KPFP, MPFP, KPF, MPF, NVFHPL                      000770   CM+FPLCTN2
C                                                                       028770   SP-RNDREL8
      PN17=URAN31(IP17)                                                 028780   SP-RNDREL9
C                                                                       028790   SP-RNDRE10
      KTEST(15,KPF)=KTEST(15,KPF)+1                                     028800   SP-RNDRE11
      IF (PN17.LE.RELIF) GO TO 100                                      028810   SP-RNDRE12
C                                                                       028820   SP-RNDRE13
      IABRT=15                                                          028830   SP-RNDRE14
      CALL ABRTTL (IABRT)                                               028840   SP-RNDRE15
C                                                                       028850   SP-RNDRE16
  100 RETURN                                                            028860   SP-RNDRE17
C                                                                       028870   SP-RNDRE18
      END                                                               028880   SP-RNDRE19

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 71 OF 82)

AD-A100 285 ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY ABERDEEN PROV--ETC F/G 19/1
COPPERHEAD OPERATIONAL PERFORMANCE EVALUATION (COPE): COMPUTER --ETC(U)
MAR 81 R S SANDMEYER
UNCLASSIFIED AMSAA-TR-318 NL

6 OF 6
AD A
100285

END
DATE
FILMED
7-81
DTIC

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
      SUBROUTINE RNGCHK                                               028900   SP-RNGCHK2
C                                                                     028910   SP-RNGCHK3
      COMMON /KTEST/ KTEST(20,6)                                      000490   CM+KTEST 2
      COMMON /RANGE/ DTRNG, MXDRNG, VISLIM, RNGNOW                    000680   CM+RANGE 2
      REAL MXDRNG                                                     000690   CM+RANGE 3
      COMMON /RPLCTN/ KEEP, NREP, KRF, NRF, NVEHKL                    000770   CM+RPLCTN2
C                                                                     028950   SP-RNGCHK7
      KTEST(3,KRF)=KTEST(3,KRF)+1                                     028960   SP-RNGCHK8
      IF (DTRNG.LE.MXDRNG) GO TO 100                                  028970   SP-RNGCHK9
C                                                                     028980   SP-RNGCH10
      IABRT=3                                                         028990   SP-RNGCH11
      CALL ABRTTL (IABRT)                                             029000   SP-RNGCH12
C                                                                     029010   SP-RNGCH13
  100 RETURN                                                          029020   SP-RNGCH14
C                                                                     029030   SP-RNGCH15
      END                                                             029040   SP-RNGCH16
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 72 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE SEPREC (A,N1,B,N2,N3)                                          029060   SP-SEPREC2
C                                                                               029070   SP-SEPREC3
      COMMON /COMENT/ CMENT(13,20), COMBUF(13), ICMNT, NTPC                     000190   CM+COMENT2
      COMMON /LOGFLG/ FIRSTL, SEQNML, SPCL(70), SHPTEC, FTSILL                  000510   CM+LOGFLG2
      LOGICAL FIRSTL, SEQNML, SPCL, SHPTEC, FTSILL                              000520   CM+LOGFLG3
      COMMON /SYMBOL/ BLANK, DECPNT, ALFBET(27), ANUMBR(11), SEP(5),            000970   CM+SYMBOL2
     1  DOLLAR                                                                  000980   CM+SYMBOL3
      DIMENSION A(N1), B(N2,N3), NCPP(10)                                       029110   SP-SEPREC7
      DIMENSION NC(80), PB(80,10)                                               029120   SP-SEPREC8
      DIMENSION CM(10)                                                          029130   SP-SEPREC9
C                                                                               029140   SP-SEPRE10
      DO 100 I=1,10                                                             029150   SP-SEPRE11
  100 CONTINUE                                                                  029160   SP-SEPRE12
      DO 110 I=1,80                                                             029170   SP-SEPRE13
  110 NC(I)=0                                                                   029180   SP-SEPRE14
      DO 120 I=1,N2                                                             029190   SP-SEPRE15
      DO 120 J=1,N3                                                             029200   SP-SEPRE16
  120 B(I,J)=BLANK                                                              029210   SP-SEPRE17
      DO 130 I=1,80                                                             029220   SP-SEPRE18
      DO 130 J=1,10                                                             029230   SP-SEPRE19
  130 PB(I,J)=BLANK                                                             029240   SP-SEPRE20
      DO 140 I=1,10                                                             029250   SP-SEPRE21
  140 NCPP(I)=0                                                                 029260   SP-SEPRE22
C                                                                               029270   SP-SEPRE23
      DO 150 I=1,N1                                                             029280   SP-SEPRE24
  150 NC(I)=IDCHAR(A,N1,I)                                                      029290   SP-SEPRE25
      DO 170 I=1,N1                                                             029300   SP-SEPRE26
      IF (NC(I).NE.5) GO TO 170                                                 029310   SP-SEPRE27
      NC(I)=3                                                                   029320   SP-SEPRE28
      DO 160 J=1,10                                                             029330   SP-SEPRE29
  160 NC(I+J)=1                                                                 029340   SP-SEPRE30
      GO TO 180                                                                 029350   SP-SEPRE31
  170 CONTINUE                                                                  029360   SP-SEPRE32
  180 CONTINUE                                                                  029370   SP-SEPRE33
C                                                                               029380   SP-SEPRE34
      IF (.NOT.FIRSTL) GO TO 220                                                029390   SP-SEPRE35
      FIRSTL=.FALSE.                                                            029400   SP-SEPRE36
      DO 190 I=73,78                                                            029410   SP-SEPRE37
      IF (NC(I).NE.0) GO TO 250                                                 029420   SP-SEPRE38
  190 CONTINUE                                                                  029430   SP-SEPRE39
      DO 200 I=63,72                                                            029440   SP-SEPRE40
      IF (NC(I).NE.3) GO TO 250                                                 029450   SP-SEPRE41
  200 CONTINUE                                                                  029460   SP-SEPRE42
      DO 210 I=79,80                                                            029470   SP-SEPRE43
      IF (NC(I).NE.3) GO TO 250                                                 029480   SP-SEPRE44
  210 CONTINUE                                                                  029490   SP-SEPRE45
      SEQNML=.TRUE.                                                             029500   SP-SEPRE46
      ENCODE (120,350,CM(1) )                                                   029510   SP-SEPRE47
      CALL COMMNT (CM,10,.FALSE.,.TRUE.)                                        029520   SP-SEPRE48
      ENCODE (120,360,CM(1) )                                                   029530   SP-SEPRE49
      CALL COMMNT (CM,10,.FALSE.,.TRUE.)                                        029540   SP-SEPRE50
      GO TO 230                                                                 029550   SP-SEPRE51
  220 IF (.NOT.SEQNML) GO TO 250                                                029560   SP-SEPRE52
  230 DO 240 I=73,78                                                            029570   SP-SEPRE53

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 73 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      NC(I)=3                                                          029580  SP-SEPRE54
  240 CONTINUE                                                         029590  SP-SEPRE55
  250 CONTINUE                                                         029600  SP-SEPRE56
C                                                                      029610  SP-SEPRE57
C  SEPARATE RECORDS ON INPUT CARD.                                     029620  SP-SEPRE58
C                                                                      029630  SP-SEPRE59
      I=1                                                              029640  SP-SEPRE60
      IW=0                                                             029650  SP-SEPRE61
      IX=0                                                             029660  SP-SEPRE62
      J=0                                                              029670  SP-SEPRE63
      LSTCMA=0                                                         029680  SP-SEPRE64
  260 IF (I.GT.N1) GO TO 320                                           029690  SP-SEPRE65
      IF (NC(I).NE.3.OR.IW.NE.IX) GO TO 280                            029700  SP-SEPRE66
  270 I=I+1                                                            029710  SP-SEPRE67
      GO TO 260                                                        029720  SP-SEPRE68
  280 IF (NC(I).EQ.3.OR.NC(I).EQ.4.OR.IW.NE.IX) GO TO 290              029730  SP-SEPRE69
      IW=IW+1                                                          029740  SP-SEPRE70
      J=0                                                              029750  SP-SEPRE71
      GO TO 310                                                        029760  SP-SEPRE72
  290 IF (NC(I).NE.4) GO TO 310                                        029770  SP-SEPRE73
      IF (I.NE.(LSTCMA+1)) GO TO 300                                   029780  SP-SEPRE74
      IW=IW+1                                                          029790  SP-SEPRE75
  300 IX=IW                                                            029800  SP-SEPRE76
      LSTCMA=I                                                         029810  SP-SEPRE77
      GO TO 270                                                        029820  SP-SEPRE78
  310 IF (NC(I).EQ.3) GO TO 270                                        029830  SP-SEPRE79
      J=J+1                                                            029840  SP-SEPRE80
      PB(J,IW)=A(I)                                                    029850  SP-SEPRE81
      NCPR(IW)=NCPR(IW)+1                                              029860  SP-SEPRE82
      GO TO 270                                                        029870  SP-SEPRE83
  320 CONTINUE                                                         029880  SP-SEPRE84
      NRE=IW                                                           029890  SP-SEPRE85
      IF (NRE.GE.0.AND.NRE.LE.10) GO TO 330                            029900  SP-SEPRE86
      STOP ' IN SEPREC: ERROR NUMBER 1 '                               029910  SP-SEPRE87
C                                                                      029920  SP-SEPRE88
C  ENCODE RECORDS INTO B ARRAY .                                       029930  SP-SEPRE89
C                                                                      029940  SP-SEPRE90
  330 CONTINUE                                                         029950  SP-SEPRE91
      DO 340 I=1,NRE                                                   029960  SP-SEPRE92
      KCHAR=NCPR(I)                                                    029970  SP-SEPRE93
      ENCODE (80,370,B(1,I) )(PB(J,I),J=1,KCHAR)                       029980  SP-SEPRE94
  340 CONTINUE                                                         029990  SP-SEPRE95
C                                                                      030000  SP-SEPRE96
      RETURN                                                           030010  SP-SEPRE97
C                                                                      030020  SP-SEPRE98
C                                                                      030030  SP-SEPRE99
C  * * *  F O R M A T   S T A T E M E N T S  * * *                     030040  SP-SEPR100
C                                                                      030050  SP-SEPR101
  350 FORMAT (92H YOU HAVE SEQUENCE NUMBERS IN COLUMNS  73-78 OF TAPE5. 030060  SP-SEPR102
     1 THEY WILL BE REMOVED BY THE PROGRAM;)                           030070  SP-SEPR103
  360 FORMAT (91H HOWEVER, IF THE TEMPORARY OPTION WAS USED TO READ IN A030080  SP-SEPR104
     1NY DATA, THE RESULTS MAY BE SUSPECT.)                            030090  SP-SEPR105
  370 FORMAT (80A1)                                                    030100  SP-SEPR106
      END                                                              030110  SP-SEPR107

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 74 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)

      SUBROUTINE SMOKE                                                  030130   SP-SMOKE 2
C                                                                       030140   SP-SMOKE 3
      COMMON /KTEST/ KTEST(20,6)                                        000490   CM*KTEST 2
      COMMON /RANDOM/ IR1, IR2, IR3, IR4, IR5, IR6, IR7, IR8, IR9, IR10,000640   CM*RANDOM2
     1  IR11, IR12, IR13, IR14, IR15, IR16, IR17, IR18, IR19, IR20, IR21000650   CM*RANDOM3
     2  , IR22, IR23, IR24, IR25, IR26, IR27, IR28                      000660   CM*RANDOM4
      COMMON /RPLCTN/ KREP, MPEP, KRF, MPF, NVEHKL                      000770   CM*RPLCTN2
      COMMON /SMOKED/ SMK2(3,3,2), SMK5(3,3,2), FPONT, NSMK2, NSMK5,    000920   CM*SMOKED2
     1  PSMKKL                                                          000930   CM*SMOKED3
      COMMON /WEATHR/ PPCLCG, PRCFLT(2), PRGCFL(11,2), W(6,11,2),       001060   CM*WEATHR2
     1  VIS(11), CLOUD(5), MCC, MVL, PASQL(6), WNDSPD(3,3), HUMID(2),   001070   CM*WEATHR3
     2  WSUM(2), WCFSUM(2), ICC, IVL                                    001080   CM*WEATHR4
C                                                                       030200   SP-SMOKE 9
      KTEST(5,KRF)=KTEST(5,KRF)+1                                       030210   SP-SMOKE10
      RN13=UPAN31(IP13)                                                 030220   SP-SMOKE11
      IF (RN13.GT.PSMKKL) GO TO 100                                     030230   SP-SMOKE12
C                                                                       030240   SP-SMOKE13
      IABRT=5                                                           030250   SP-SMOKE14
      CALL ABRTTL (IABRT)                                               030260   SP-SMOKE15
C                                                                       030270   SP-SMOKE16
  100 RETURN                                                            030280   SP-SMOKE17
C                                                                       030290   SP-SMOKE18
      END                                                               030300   SP-SMOKE19

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 75 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE SMPLCD (A,ND1,ND2,NRS,NDR,IRNDM,XSMPL,ICLASS)          030320  SP-SMPLC02
C                                                                       030330  SP-SMPLC03
      DIMENSION A(ND1,ND2)                                              030340  SP-SMPLC04
C                                                                       030350  SP-SMPLC05
      ICLASS=0                                                          030360  SP-SMPLC06
      RN=URAN31(IRNDM)                                                  030370  SP-SMPLC07
      IF (RN.LT.1.0) GO TO 100                                          030380  SP-SMPLC08
      WRITE (6,130) RN                                                  030390  SP-SMPLC09
      CALL STRACE                                                       030400  SP-SMPLC10
      STOP ' IN SMPLCD: RANDOM NUMBER TOO LARGE '                       030410  SP-SMPLC11
C                                                                       030420  SP-SMPLC12
  100 CONTINUE                                                          030430  SP-SMPLC13
      NDRM1=NDR-1                                                       030440  SP-SMPLC14
      DO 110 I=1,NDRM1                                                  030450  SP-SMPLC15
      IF (A(I,1).LE.RN.AND.RN.LT.A(I+1,1)) GO TO 120                    030460  SP-SMPLC16
  110 CONTINUE                                                          030470  SP-SMPLC17
C                                                                       030480  SP-SMPLC18
      CALL STRACE                                                       030490  SP-SMPLC19
      STOP ' IN SMPLCD: FAILED TO BRACKET RANDOM NUMBER '               030500  SP-SMPLC20
C                                                                       030510  SP-SMPLC21
  120 XSMPL=A(I,NRS)+(RN-A(I,1))*(A(I+1,NRS)-A(I,NRS))/(A(I+1,1)-A(I,1))030520  SP-SMPLC22
      ICLASS=I                                                          030530  SP-SMPLC23
C                                                                       030540  SP-SMPLC24
      RETURN                                                            030550  SP-SMPLC25
C                                                                       030560  SP-SMPLC26
C                                                                       030570  SP-SMPLC27
C  * * *   F O R M A T   S T A T E M E N T S   * * *                    030580  SP-SMPLC28
C                                                                       030590  SP-SMPLC29
  130 FORMAT (1H ,59H IN SMPLCD WITH RANDOM NUMBER GREATER THAN 1.0,    030600  SP-SMPLC30
     1   RN = ,F10.7)                                                   030610  SP-SMPLC31
      END                                                               030620  SP-SMPLC32

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 76 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE TIMCHK                                                030640   SP-TIMCHK2
C                                                                      030650   SP-TIMCHK3
      COMMON /FLAG/ ENDPEP, IDN, IPPEPL, IDIGTL, IPETY, IPNDOC, IT3FLG, 000300   CM+FLAG  2
     1  ENDFLG                                                         000310   CM+FLAG  3
      LOGICAL ENDFLG, ENDPEP                                           000320   CM+FLAG  4
      COMMON /KTEST/ KTEST(20,6)                                       000490   CM+KTEST 2
      COMMON /RPLCTN/ KPEP, NPEP, KPF, NPF, NVEHKL                     000770   CM+RPLCTN2
      COMMON /TIME/ DUPLOS, TIMNOW, TOF, TBP, TIMFA, TIMRAI            001020   CM+TIME  2
C                                                                      030700   SP-TIMCHK8
      KTEST(11,KPF)=KTEST(11,KPF)+1                                    030710   SP-TIMCHK9
      IF (TIMNOW.LT.DUPLOS) GO TO 100                                  030720   SP-TIMCH10
C                                                                      030730   SP-TIMCH11
      IABPT=11                                                         030740   SP-TIMCH12
      CALL ABRTTL (IABPT)                                              030750   SP-TIMCH13
C                                                                      030760   SP-TIMCH14
  100 RETURN                                                           030770   SP-TIMCH15
C                                                                      030780   SP-TIMCH16
      END                                                              030790   SP-TIMCH17

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 77 OF 82)

```
0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE TITLE                                                        030810  SR-TITLE 2
C                                                                             030820  SR-TITLE 3
      COMMON /RUNDAT/ DAT, TIM, NCASE, ICCPFG, IPPFG                          000840  CM+RUNDAT2
      COMMON /STITLE/ SLVERS, VERDAT(2)                                       000950  CM+STITLE2
C                                                                             030850  SR-TITLE 6
      CALL DATE (DAT)                                                         030860  SR-TITLE 7
      CALL TIME (TIM)                                                         030870  SR-TITLE 8
C                                                                             030880  SR-TITLE 9
      WRITE (6,150)                                                           030890  SR-TITLE10
      WRITE (6,160)                                                           030900  SR-TITLE11
      DO 100 I=1,4                                                            030910  SR-TITLE12
      WRITE (6,170)                                                           030920  SR-TITLE13
  100 CONTINUE                                                                030930  SR-TITLE14
      WRITE (6,180)                                                           030940  SR-TITLE15
      DO 110 I=1,8                                                            030950  SR-TITLE16
      WRITE (6,170)                                                           030960  SR-TITLE17
  110 CONTINUE                                                                030970  SR-TITLE18
      WRITE (6,190)                                                           030980  SR-TITLE19
      WRITE (6,170)                                                           030990  SR-TITLE20
      WRITE (6,200)                                                           031000  SR-TITLE21
      WRITE (6,170)                                                           031010  SR-TITLE22
      WRITE (6,210)                                                           031020  SR-TITLE23
      WRITE (6,170)                                                           031030  SR-TITLE24
      WRITE (6,220)                                                           031040  SR-TITLE25
      WRITE (6,170)                                                           031050  SR-TITLE26
      WRITE (6,230)                                                           031060  SR-TITLE27
      WRITE (6,170)                                                           031070  SR-TITLE28
      WRITE (6,240)                                                           031080  SR-TITLE29
      WRITE (6,170)                                                           031090  SR-TITLE30
      WRITE (6,250)                                                           031100  SR-TITLE31
      DO 120 I=1,11                                                           031110  SR-TITLE32
      WRITE (6,170)                                                           031120  SR-TITLE33
  120 CONTINUE                                                                031130  SR-TITLE34
      WRITE (6,260)                                                           031140  SR-TITLE35
      WRITE (6,170)                                                           031150  SR-TITLE36
      WRITE (6,270)                                                           031160  SR-TITLE37
      WRITE (6,170)                                                           031170  SR-TITLE38
      WRITE (6,280)                                                           031180  SR-TITLE39
      WRITE (6,170)                                                           031190  SR-TITLE40
      WRITE (6,290)                                                           031200  SR-TITLE41
      DO 130 I=1,5                                                            031210  SR-TITLE42
      WRITE (6,170)                                                           031220  SR-TITLE43
  130 CONTINUE                                                                031230  SR-TITLE44
      WRITE (6,300) SLVERS,VERDAT(1),VERDAT(2)                                031240  SR-TITLE45
      WRITE (6,170)                                                           031250  SR-TITLE46
      WRITE (6,310) DAT,TIM                                                   031260  SR-TITLE47
      DO 140 I=1,5                                                            031270  SR-TITLE48
      WRITE (6,170)                                                           031280  SR-TITLE49
  140 CONTINUE                                                                031290  SR-TITLE50
      WRITE (6,160)                                                           031300  SR-TITLE51
      RETURN                                                                  031310  SR-TITLE52
C                                                                             031320  SR-TITLE53
C                                                                             031330  SR-TITLE54
C  * * *  F O R M A T   S T A T E M E N T S  * * *                            031340  SR-TITLE55

0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 78 Of 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)

C                                                                          031350  SP-TITLE56
  150 FORMAT (1H1)                                                         031360  SP-TITLE57
  160 FORMAT (1H ,13(10H**********))                                       031370  SP-TITLE58
  170 FORMAT (1H ,1H*,128X,1H*)                                            031380  SP-TITLE59
  180 FORMAT (1H ,1H*,19X,89HC O P P E R H E A D   O P E R A T I O N A L031390  SP-TITLE60
     1   P E R F O R M A N C E   E V A L U A T I O N,20X,1H*)              031400  SP-TITLE61
  190 FORMAT (1H ,1H*,32X,29H-PROGRAM BY RICHARD SANDMEYER,67X,1H*)        031410  SP-TITLE62
  200 FORMAT (1H ,1H*,32X,42H-MODEL DEVELOPMENT AND DATA COLLECTION BY:,031420  SP-TITLE63
     154X,1H*)                                                             031430  SP-TITLE64
  210 FORMAT (1H ,1H*,34X,14HDAVID BARNHART,8X,15HJULIAN CHERNICK,5X,15H031440  SP-TITLE65
     1DIANA FREDERICK,37X,1H*)                                             031450  SP-TITLE66
  220 FORMAT (1H ,1H*,34X,17HRICHARD SANDMEYER,5X,15HRICHARD SCUNGIO,5X,031460  SP-TITLE67
     114HMICHAEL STARKS,38X,1H*)                                           031470  SP-TITLE68
  230 FORMAT (1H ,1H*,34X,13HEDWARD STAUCH,9X,11HANNIE YOUNG,61X,1H*)      031480  SP-TITLE69
  240 FORMAT (1H ,1H*,32X,38H-BASED ON A PROPOSAL BY HERBERT FALLIN,58X,031490  SP-TITLE70
     11H*)                                                                 031500  SP-TITLE71
  250 FORMAT (1H ,1H*,32X,48H-DONE AT THE REQUEST OF DAVID HARDISON (DUS031510  SP-TITLE72
     1A-OR),48X,1H*)                                                       031520  SP-TITLE73
  260 FORMAT (1H ,1H*,36X,53HUNITED STATES ARMY MATERIEL SYSTEMS ANALYSI031530  SP-TITLE74
     1S ACTIVITY,39X,1H*)                                                  031540  SP-TITLE75
  270 FORMAT (1H ,1H*,51X,23HGROUND WARFARE DIVISION,54X,1H*)              031550  SP-TITLE76
  280 FORMAT (1H ,1H*,47X,31HSUPPORT WEAPONS ANALYSIS BRANCH,50X,1H*)      031560  SR-TITLE77
  290 FORMAT (1H ,1H*,50X,26HSYSTEMS EVALUATION SECTION,52X,1H*)           031570  SP-TITLE78
  300 FORMAT (1H ,1H*,40X,13HCOPE VERSION ,F3.1,15H CURRENT AS OF ,2A10,031580  SP-TITLE79
     137X,1H*)                                                             031590  SP-TITLE80
  310 FORMAT (1H ,1H*,42X,16HTHIS RUN DONE ON,A10,7HAT TIME,A10,43X,1H*)031600  SP-TITLE81
      END                                                                  031610  SP-TITLE82

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888889 COLUMN NUMBERS
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 79 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE USET                                                   031630   SP-USET  2
C                                                                       031640   SP-USET  3
      COMMON /DVALUE/ D(70), DPPT, DDT, DIDN, DIDIGT, DSMK(2)           000280   CM+DVALUE2
      COMMON /LOGFLG/ FIRSTL, SEQNML, SPCL(70), SHPTEC, FTSILL          000510   CM+LOGFLG2
      LOGICAL FIRSTL, SEQNML, SPCL, SHPTEC, FTSILL                      000520   CM+LOGFLG3
      COMMON /OVER/ IOVER(70)                                           000560   CM+OVER  2
      COMMON /UVALUE/ U(70), UPPT, UDT, UIDN, UIDIGT, USMK(2)           001040   CM+UVALUE2
C                                                                       031690   SP-USET  8
      DO 100 I=1,70                                                     031700   SP-USET  9
      U(I)=D(I)                                                         031710   SP-USET 10
      IOVER(I)=0                                                        031720   SP-USET 11
      SPCL(I)=.FALSE.                                                   031730   SP-USET 12
  100 CONTINUE                                                          031740   SP-USET 13
C                                                                       031750   SP-USET 14
      UPPT=DPPT                                                         031760   SP-USET 15
      UDT=DDT                                                           031770   SP-USET 16
      UIDN=DIDN                                                         031780   SP-USET 17
      UIDIGT=DIDIGT                                                     031790   SP-USET 18
      USMK(1)=DSMK(1)                                                   031800   SP-USET 19
      USMK(2)=DSMK(2)                                                   031810   SP-USET 20
      FTSILL=.TRUE.                                                     031820   SP-USET 21
C                                                                       031830   SP-USET 22
      RETURN                                                            031840   SP-USET 23
      END                                                               031850   SP-USET 24

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1  FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 80 OF 82)

```
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE VISCHK                                                 031870  SP-VISCHK2
C                                                                       031880  SP-VISCHK3
      COMMON /KTEST/ KTEST(20,6)                                        000490  CM+KTEST 2
      COMMON /RANGE/ DTRNG, MXDRNG, VISLIM, RNGNOW                      000680  CM+RANGE 2
      REAL MXDRNG                                                       000690  CM+RANGE 3
      COMMON /RPLCTN/ KREP, NREP, KRF, NRF, NVEHKL                      000770  CM+RPLCTN2
C                                                                       031920  SP-VISCHK7
      KTEST(2,KRF)=KTEST(2,KRF)+1                                       031930  SP-VISCHK8
      IF (DTRNG.LE.1000.*VISLIM) GO TO 100                              031940  SP-VISCHK9
C                                                                       031950  SP-VISCH10
      IABRT=2                                                           031960  SP-VISCH11
      CALL ABRTTL (IABRT)                                               031970  SP-VISCH12
C                                                                       031980  SP-VISCH13
  100 RETURN                                                            031990  SP-VISCH14
C                                                                       032000  SP-VISCH15
      END                                                               032010  SP-VISCH16

000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D

(PAGE 81 OF 82)

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)

      SUBROUTINE WARNDD                                                  032030   SP-WARND02
C                                                                        032040   SP-WARND03
      COMMON /FLAG/ ENDREP, IDM, IPREPL, IDIGTL, IPRTY, IPNDOC, IT3FLG,  000300   CM+FLAG  2
     1 ENDFLG                                                            000310   CM+FLAG  3
      LOGICAL ENDFLG, ENDREP                                             000320   CM+FLAG  4
      COMMON /KTEST/ KTEST(20,6)                                         000490   CM+KTEST 2
      COMMON /RANDOM/ IP1, IP2, IP3, IP4, IP5, IP6, IP7, IR8, IP9, IP10, 000640   CM+RANDOM2
     1  IR11, IP12, IP13, IP14, IP15, IP16, IP17, IP18, IP19, IP20, IR21 000650   CM+RANDOM3
     2  , IR22, IR23, IR24, IP25, IP26, IP27, IP28                       000660   CM+RANDOM4
      COMMON /RPLCTN/ KREP, NREP, KRF, NRF, NVEHKL                       000770   CM+RPLCTN2
      COMMON /RSPTIM/ RATRTM(3), XMVADF(2), DUM1, RCSPTM(2,2), XMTTIM(2, 000790   CM+RSPTIM2
     1  2), TMEAN(2,2), TSIGMA(2,2), DETTMA(10,2), MDT, TRAPRY(3,2),     000800   CM+RSPTIM3
     2  PRSPTM, TRBAR, PRSDT, PRSVT, ALPHA(2,2), BETA(2,2), DETTIM,      000810   CM+RSPTIM4
     3  IDSGTP, RUMC, PCRMSG, PRDOW                                      000820   CM+RSPTIM5
      COMMON /TIME/ DURLOS, TIMNOW, TOF, TRP, TIMRA, TIMRA1              001020   CM+TIME  2
C                                                                        032110   SR-WARND10
      KTEST(13,KRF)=KTEST(13,KRF)+1                                      032120   SR-WARND11
      IF (KRF.GT.1) GO TO 100                                            032130   SP-WARND12
      TIMRA1=TIMRA                                                       032140   SP-WARND13
      PN16=UPAN31(IR16)                                                  032150   SP-WARND14
      IF (PN16.LE.PRDOW) GO TO 100                                       032160   SR-WARND15
C                                                                        032170   SR-WARND16
      IABRT=13                                                           032180   SR-WARND17
      CALL ABRTTL (IABRT)                                                032190   SP-WARND18
C                                                                        032200   SR-WARND19
  100 RETURN                                                             032210   SP-WARND20
C                                                                        032220   SR-WARND21
      END                                                                032230   SP-WARND22

00000000011111111112222222222333333333344444444445555555555666666666677777777778888888888889 COLUMN NUMBERS
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890 (READ VERTICALLY)
```

LISTING D-1 FORTRAN LISTING OF COPE PROGRAM - CONT'D
(PAGE 82 OF 82)

```
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

FIRST CASE
$$$ THIS COMMENT WILL APPEAR ON THE $$$ COMMENT PAGE
WEATHER,JUNE,0600
ACQRNGDIST,CLOSE
PLOT,PRINTER,ON
END
NEXT CASE,DP
DESIGTYPE,MULE
RESPTIME,PARAM,150,N
CONTROL,ECHO,LONG
PLOT,PRINTER,OFF
$ THE OVERRIDE CARD WHICH FOLLOWS OVERRIDES THE NORMAL CHOICE OF
$ PE DATA WITH THE PE DATA RECORD NAMED PEOOAJAAJS
$  NOTE THE USE OF THE $ BEFORE THE RECORD NAME
OVERRIDE,PEDATA,$PEOOAJAAJS
END
NEXT CASE,RD
W,D,2200
LOSMODE,RD
$ THE NEXT CARD SIGNALS FORMATTED DATA IS TO BE READ INTO RANDOM
$ OCCURRENCE DATA BLOCK
TEMP,RODIST
     5
   15.0
 1000.0    1500.0     2000.0     3000.0     7000.0
     .6        .5         .4         .3         .2
$$ THIS COMMENT WILL BE PRINTED ON CASE 3 HEADING PAGE
TARGET TYPE,4
DIFFSUPPR,NONE
RUNC,200
SMOKE,SMKRNDS,200,400
TRP,10
NUMVEHTGT,6
DISTRVEH,50
END
NEXT CASE ,RD
REFLECTIVITY,.3
SMOKE,PERSMOKE,.15
DUST,.20
PROBDOWAEN,.8
ANGLE T,60
ENDF

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000111111111111111111111 COLUMN NOS.
000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000000000011111111112
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LISTING D-2  SAMPLE INPUT FOR COPE

```
JOB CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
ACCOUNT CARD.  CHANGE THIS CARD TO CONFORM TO INSTALLATION USAGE.
COMMENT.  THE NEXT CARD DECLARES TAPE5 TO HAVE RECORD TYPE Z AND 80 CHARACTERS
COMMENT.  PER LINE(RECORD).
FILE,TAPE5,RT=Z,FL=80,BT=C.
COMMENT.  THE NEXT STATEMENT OBTAINS A COPY OF THE COPE INPUT AND PUTS IT IN
COMMENT.  A LOCAL FILE NAMED TAPE5.
GETPF,TAPE5,COPESAMPLEINPUT,ID=SANDMEYER,ST=MFA
COMMENT.  THE NEXT STATEMENT CAUSES TAPE5 (COPE INPUT) TO BE LISTED.  THIS IS
COMMENT.  OFTEN A WISE PRECAUTION FOR CHECKING PURPOSES.
COPYSPF,TAPE5.
REWIND,TAPE5.
COMMENT.  THE NEXT STATEMENT ATTACHES THE PREVIOUSLY COMPILED (IN ANOTHER JOB)
COMMENT.  OBJECT CODE FOR THE COPE PROGRAM.
ATTACH,LGO,COPESAMPLEBINARY,ID=SANDMEYER.
COMMENT.  THE NEXT STATEMENT ATTACHES THE COPE "DATA BASE" FILE (TAPE11) WHICH
COMMENT.  WAS CREATED AND MODIFIED BY VARIOUS PREPROCESSOR RUNS (RAM & PREPNS)
ATTACH,TAPE11,SAMPLETAPE11,ID=SANDMEYER.
COMMENT.  THE NEXT TWO CARDS ATTACH THE INTERNATIONAL MATHEMATICAL AND
COMMENT.  STATISTICAL LIBRARY (IMSL) AND DECLARE IT TO BE A LIBRARY TO BE
COMMENT.  USED IN SATISFYING EXTERNAL REFERENCES IN THE COPE PROGRAM.
ATTACH,IMSL.
LIBRARY,IMSL.
COMMENT.  THE NEXT CARD DECLARES TAPE8 TO HAVE RECORD TYPE Z AND 140 CHARACTERS
COMMENT.  PER LINE.  THIS IS DONE BECAUSE TAPE8 IS TO BE SAVED ON THE "FRONT
COMMENT.  END" COMPUTER WHERE RECORD TYPE Z IS THE DEFAULT RECORD TYPE.
FILE,TAPE8,RT=Z,FL=140,BT=C.
COMMENT.  THE NEXT STATEMENT LOADS AND EXECUTES THE COPE PROGRAM USING TAPE5 AS
COMMENT.  INPUT.
LGO(TAPE5)
COMMENT.  THE NEXT STATEMENT SAVES A COPY OF TAPE8 (COPE AUXILIARY OUTPUT)
COMMENT.  ON THE "FRONT END" COMPUTER.  THIS TAPE8 IS USED TO PRODUCE THE
COMMENT.  LISTINGS OF SECTION 7.6 OF THE COPE DOCUMENTATION.
SAVEPF,TAPE8,COPEAUXOUTPUT,ID=SANDMEYER,ST=MFA.
COMMENT. END OF JOB
```

LISTING D-3  SAMPLE RUNSTREAM FOR COPE

```
FIRST CASE
$$$ THIS COMMENT WILL APPEAR ON THE $$$ COMMENT PAGE
WEATHER,JUNE,0600
ACQRNGDIST,CLOSE
PLOT,PRINTER,ON
END
NEXT CASE,DR
DESIGTYPE,MULE
RESPTIME,PARAM,150,N
CONTROL,ECHO,LONG
PLOT,PRINTER,OFF
$ THE OVERRIDE CARD WHICH FOLLOWS OVERRIDES THE NORMAL CHOICE OF
$ PE DATA WITH THE PE DATA RECORD NAMED PEOOA)AAJS
$  NOTE THE USE OF THE $ BEFORE THE RECORD NAME
OVERRIDE,PEDATA,$PEOOA)AAJS
END
NEXT CASE,PD
W,D,2200
LOSMODE,RO
$ THE NEXT CARD SIGNALS FORMATTED DATA IS TO BE READ INTO RANDOM
$ OCCURRENCE DATA BLOCK
TEMP,RODIST
    5
   15.0
 1000.0    1500.0    2000.0    3000.0    7000.0
     .6        .5        .4        .3        .2
$$ THIS COMMENT WILL BE PRINTED ON CASE 3 HEADING PAGE
TARGET TYPE,4
DFIRESUPPP,NONE
PUMC,200
SMOKE,SMKRNDS,200,400
TBP,10
NUMVEHTGT,6
DISTBVEH,50
END
NEXT CASE ,PD
REFLECTIVITY,.3
SMOKE,PPBSMOKE,.15
DUST,.20
PROBDOVAPH,.8
ANGLE T,60
ENDF
```

LISTING D-4  SAMPLE OUTPUT FROM COPE

(PAGE 1 OF 11)

```
C O P P E R H E A D   O P E R A T I O N A L   P E R F O R M A N C E   E V A L U A T I O N

  -PROGRAM BY RICHARD SANDMEYER
  -MODEL DEVELOPMENT AND DATA COLLECTION BY:
     DAVID BARNHART        JULIAN CHERNICK      DIANA FREDERICK
     RICHARD SANDMEYER     RICHARD SCUNGIO      MICHAEL STARKS
     EDWARD STAUCH         ANNIE YOUNG
  -BASED ON A PROPOSAL BY HERBERT FALLIN
  -DONE AT THE REQUEST OF DAVID HARDISON (DUSA-OR)

       UNITED STATES ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY
                    GROUND WARFARE DIVISION
                SUPPORT WEAPONS ANALYSIS BRANCH
                  SYSTEMS EVALUATION SECTION

          COPE VERSION 6.2 CURRENT AS OF 6 MARCH 1980
            THIS RUN DONE ON 08/14/80 AT TIME 15.41.16.
```

LISTING D-4 SAMPLE OUTPUT FROM COPE

(PAGE 2 OF 11)

```
$$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$     COMMENTS FOR THIS SET OF CASES     $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$ $$$
** USER **  THIS COMMENT WILL APPEAR ON THE $$$ COMMENT PAGE
```

LISTING D-4  SAMPLE OUTPUT FROM COPE

(PAGE 3 OF 11)

```
* * * * * * * * * * * * * * * * * * * * * * * * *   C A S E   N U M B E R   1 * * * * * * * * * * * * * * * * * * * * * * * * *

NUMBER OF REPLICATIONS=10000

TARGET
TYPE 1
NUMBER OF VEHICLES=10
MEAN DISTANCE BETWEEN VEHICLES= 61.M
TARGET VELOCITY=3. M/SEC
TARGET HEADING= 0. DEGREES
TARGET'S REFLECTIVITY= .10

FIRING INFO
NUMBER OF ROUNDS PER ENGAGEMENT= 6
ROUND RELIABILITY= .96
TIME BETWEEN ROUNDS= 20. SEC
TIME OF FLIGHT= 31.0 SEC
NOMINAL GUN-TARGET RANGE= 8. KM
ANGLE T= 25. DEGREES
DEFLECTION BIAS= 0. METERS

MISC
DELTA T3 ASSUMED TO BE ZERO
SEEKER SENSITIVITY= 60. JOULES/M**2

DETECT, COMMO, AND PROC TIMES
PRIORITY PRE-PLANNED TARGET
DIGITAL COMMO
DAY TIME
NOMINAL RESPONSE TIME=106. SEC
PROB SUCCESSFUL DIGITAL TRAN= .915
PROB CORRECT MESSAGE = .975
PROB DO WARNED TO LASE = .975

DESIGNATOR
TYPE 1
MAXIMUM DESIGNATOR RANGE=7000. M
BAIL-OUT RANGE=1000. M
DESIGNATOR AT VANTAGE POINT

DESIGNATOR SUPPRESSED
PROB SUPPRESSED BY INDIRECT FIRE=.01
PROB SUPPRESSED BY DIRECT FIRE: HIGH

WEATHER
MONTH OF JUN
TIME OF DAY: 0600

TERRAIN AND LOS
CLOSE TERRAIN
CONTINUOUS LOS ("SHOOTING GALLERY")

OBSCURANTS
PROB SMOKE KILLS MISSION= .75
VOLUME OF SMOKE FIRED: 484 + 846
PROB DUST KILLS MISSION= .37

$$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$   C O M M E N T S   $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$

** COPE **   IN PPLOT FOR CASE  1 WITH IPREG= 1 .  NO PLOT PRODUCED .
```

LISTING D-4 SAMPLE OUTPUT FROM COPE

(PAGE 4 OF 11)

CAUSE OF ABORT

| | | | | |
|---|---|---|---|---|
| | | | ***** OCCASIONS (10000) ***** | |
| 1 | DESIGNATOR IN INDIRECT FIRE | 84(0.84%) | NUMBER TESTED (% PASSED): | 10000(99.16%) |
| 2 | TARGET BEYOND VISIBILITY RANGE | 415(4.15%) | NUMBER TESTED (% PASSED): | 9916(95.81%) |
| 3 | TARGET BEYOND DESIGNATOR RANGE | 0(0.00%) | NUMBER TESTED (% PASSED): | 9501(100.0%) |
| 4 | TARGET NOT DETECTED | 0(0.00%) | NUMBER TESTED (% PASSED): | 9501(100.0%) |
| 5 | SMOKE KILLED MISSION | 7123(71.23%) | NUMBER TESTED (% PASSED): | 9501(25.03%) |
| 6 | DUST KILLED MISSION | 887(8.87%) | NUMBER TESTED (% PASSED): | 2378(62.70%) |
| 7 | DESIGNATOR BAILED-OUT PRE-COMM | 484(4.84%) | NUMBER TESTED (% PASSED): | 1491(67.54%) |
| | | | ***** ATTEMPTED ENGAGEMENTS ( 1007) ***** | |
| 8 | COMMO OUT (BOTH VOICE & DIGTL) | 70(0.70%) | NUMBER TESTED (% PASSED): | 1007(93.05%) |
| 9 | ERROR IN TRANSMISSION | 21(0.21%) | NUMBER TESTED (% PASSED): | 937(97.76%) |
| 10 | DESIGNATOR FAILED-OUT POST-COM | 159(1.59%) | NUMBER TESTED (% PASSED): | 916(82.64%) |
| 11 | LOS LOST - NO FIRING | 55(0.55%) | NUMBER TESTED (% PASSED): | 757(92.73%) |

***** SHOTS ( 702) *****

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | ROUND NUMBER : | 1 | 2 | 3 | 4 | 5 | 6 |
| 12 | LOS LOST DURING MISSION | 30(0.30%) | 62(0.62%) | 96(0.96%) | 151(1.51%) | 242(2.42%) | 345(3.45%) |
| | NUMBER TESTED (% PASSED) | 702(95.73%) | 702(91.17%) | 702(86.32%) | 702(78.49%) | 702(65.53%) | 702(50.85%) |
| 13 | DESIGNATOR NOT WARNED IN TIME | 15(0.15%) | 0(0.00%) | 0(0.00%) | 0(0.00%) | 0(0.00%) | 0(0.00%) |
| | NUMBER TESTED (% PASSED) | 672(97.77%) | 640(100.0%) | 606(100.0%) | 551(100.0%) | 460(100.0%) | 357(100.0%) |
| 14 | DESIGNATOR IN DIRECT FIRE | 176(1.76%) | 173(1.73%) | 194(1.94%) | 183(1.83%) | 164(1.64%) | 144(1.44%) |
| | NUMBER TESTED (% PASSED) | 657(73.21%) | 640(72.97%) | 606(67.99%) | 551(66.79%) | 460(64.35%) | 357(59.66%) |
| 15 | ROUND RELIABILITY FAILURE | 20(0.20%) | 16(0.16%) | 15(0.15%) | 16(0.16%) | 9(0.09%) | 13(0.13%) |
| | NUMBER TESTED (% PASSED) | 481(95.84%) | 467(96.57%) | 412(96.36%) | 368(95.65%) | 296(96.96%) | 213(93.90%) |
| 16 | TARGET OBSCURED BY MINI-TERRN | 54(0.54%) | 61(0.61%) | 83(0.83%) | 55(0.55%) | 46(0.46%) | 29(0.29%) |
| | NUMBER TESTED (% PASSED) | 461(88.29%) | 451(86.47%) | 397(79.09%) | 352(84.38%) | 287(83.97%) | 200(85.50%) |
| 17 | ROUND DID NOT ENGAGE TARGET | 1(0.01%) | 1(0.01%) | 4(0.04%) | 2(0.02%) | 0(0.00%) | 0(0.00%) |
| | NUMBER TESTED (% PASSED) | 407(99.75%) | 390(99.74%) | 314(98.73%) | 297(99.33%) | 241(100.0%) | 171(100.0%) |
| 18 | ROUND ENGAGED BUT DID NOT HIT | 105(1.05%) | 83(0.83%) | 74(0.74%) | 54(0.54%) | 43(0.43%) | 49(0.49%) |
| | NUMBER TESTED (% PASSED) | 406(74.14%) | 389(78.66%) | 310(76.13%) | 295(81.59%) | 241(82.16%) | 171(71.35%) |
| 19 | ROUND HIT BUT DID NOT KILL | 156(1.56%) | 159(1.59%) | 104(1.04%) | 128(1.28%) | 96(0.96%) | 64(0.64%) |
| | NUMBER TESTED (% PASSED) | 301(48.17%) | 306(48.04%) | 236(55.93%) | 241(46.89%) | 198(51.52%) | 122(47.54%) |
| | ***** KILLS | 145 | 147 | 132 | 113 | 102 | 58 ***** |
| | PROB(KILL/DCSN)= | 1.45% | 1.47% | 1.32% | 1.13% | 1.02% | .58% |
| | PROB(KILL/ATMP ENG)= | 14.40% | 14.60% | 13.11% | 11.22% | 10.13% | 5.76% |
| | PROB(KILL/SHOT)= | 20.66% | 20.94% | 18.80% | 16.10% | 14.53% | 8.26% |
| | PROB(LOS)= | 88.72% | 84.54% | 80.05% | 72.79% | 60.77% | 47.16% |

PROB(SHOT/DCSN)= 7.02%     PROB(SHOT/ATMP ENG)= 69.71%     PROB(ATMP ENG/DCSN)= 10.07%

DESIGNATOR SURVIVABILITY: CASES IN WHICH DESIGNATOR KILLED - 141(1.41%)     NUMBER TESTED (% NOT KILLED) - 657(78.54%)

LISTING D-4 SAMPLE OUTPUT FROM COPE

(PAGE 5 OF 11)

```
* * * * * * * * * * * * * * * * * * * * * * * * *   C A S E   N U M B E R   2   * * * * * * * * * * * * * * * * * * * * * * * * *

NUMBER OF REPLICATIONS=10000

TARGET
  TYPE  1
  NUMBER OF VEHICLES=10
  MEAN DISTANCE BETWEEN VEHICLES= 61.M
  TARGET VELOCITY=2. M/SEC
  TARGET HEADING=  0. DEGREES
  TARGET'S REFLECTIVITY= .10

FIRING INFO
  NUMBER OF ROUNDS PER ENGAGEMENT= 6
  ROUND RELIABILITY= .96
  TIME BETWEEN ROUNDS= 20. SEC
  TIME OF FLIGHT= 31.0 SEC
  NOMINAL GUN-TARGET RANGE=  8. KM
  ANGLE T= 25. DEGREES
  DEFLECTION BIAS=   0. METERS

MISC
  DELTA T3 ASSUMED TO BE ZERO
  SEEKER SENSITIVITY= 60. JOULES/M**2

DETECT, COMMO, AND PROC TIMES
  PARAMETERIZED RESPONSE TIME=150. SEC
  PRIORITY PRE-PLANNED TARGET
  DIGITAL COMMO
  NIGHT TIME
  NOMINAL RESPONSE TIME=181. SEC
  PROB SUCCESSFUL DIGITAL TRAN= .915
  PROB CORRECT MESSAGE =  .975
  PROB DO WARNED TO LASE =  .975

DESIGNATOR
  TYPE  2
  MAXIMUM DESIGNATOR RANGE=2600. M
  BAIL-OUT RANGE=1000. M
  DESIGNATOR AT VANTAGE POINT

DESIGNATOR SUPPPESSED
  PROB SUPPRESSED BY INDIRECT FIRE=.01
  PROB SUPPRESSED BY DIRECT FIRE: HIGH

WEATHER
  MONTH OF JUN
  TIME OF DAY: 0600

TERRAIN AND LOS
  CLOSE TERRAIN
  CONTINUOUS LOS ("SHOOTING GALLERY")

OBSCURANTS
  PROB SMOKE KILLS MISSION= .75
  VOLUME OF SMOKE FIRED:  484 +  846
  PROB DUST KILLS MISSION= .37

$$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$      C O M M E N T S      $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$

** COPE **  CHOICE OF NIGHT TIME FOR RESPONSE TIME DATA IS INCONSISTENT WITH TIME OF 0600 FOR WEATHER DATA
** COPE **  RECORD NAMED: PEOOAIAAJS USED TO OVERRIDE FOR PEDATA
** COPE **  PEDATA FOR: TP=106. DT=1 VEL=5. GTR= 8. REFL= .10 ANGLET= 25. DEFL=  0. TGTHD= 0. SKSEN=60.
```

LISTING D-4  SAMPLE OUTPUT FROM COPE

(PAGE 6 OF 11)

```
          CAUSE OF ABORT
                                             ***** OCCASIONS (10000) *****

 1   DESIGNATOR IN INDIRECT FIRE          84(0.84%)    NUMBER TESTED (% PASSED):  10000(99.16%)

 2   TARGET BEYOND VISIBILITY RANGE      415(4.15%)    NUMBER TESTED (% PASSED):   9916(95.81%)

 3   TARGET BEYOND DESIGNATOR RANGE     2460(24.60%)   NUMBER TESTED (% PASSED):   9501(74.11%)

 4   TARGET NOT DETECTED                   0(0.00%)    NUMBER TESTED (% PASSED):   7041(100.0%)

 5   SMOKE KILLED MISSION               5254(52.54%)   NUMBER TESTED (% PASSED):   7041(25.38%)

 6   DUST KILLED MISSION                 665(6.65%)    NUMBER TESTED (% PASSED):   1787(62.79%)

 7   DESIGNATOR BAILED-OUT PRE-COMM      554(5.54%)    NUMBER TESTED (% PASSED):   1122(50.62%)

                                        ***** ATTEMPTED ENGAGEMENTS (  568) *****

 8   COMMO OUT (BOTH VOICE & DIGTL)       39(0.39%)    NUMBER TESTED (% PASSED):    568(93.13%)

 9   ERROR IN TRANSMISSION                14(0.14%)    NUMBER TESTED (% PASSED):    529(97.35%)

10   DESIGNATOR FAILED-OUT POST-COM       87(0.87%)    NUMBER TESTED (% PASSED):    515(83.11%)

11   LOS LOST - NO FIRING                  0(0.00%)    NUMBER TESTED (% PASSED):    428(100.0%)

                                             ***** SHOTS (  428) *****
       ROUND NUMBER :                      1             2             3             4             5             6

12   LOS LOST DURING MISSION            0(0.00%)      0(0.00%)      0(0.00%)      0(0.00%)      0(0.00%)      0(0.00%)
       NUMBER TESTED (% PASSED)       428(100.0%)   428(100.0%)   428(100.0%)   428(100.0%)   428(100.0%)   428(100.0%)

13   DESIGNATOR NOT WARNED IN TIME     10(0.10%)      0(0.00%)      0(0.00%)      0(0.00%)      0(0.00%)      0(0.00%)
       NUMBER TESTED (% PASSED)       428(97.66%)   428(100.0%)   428(100.0%)   428(100.0%)   428(100.0%)   428(100.0%)

14   DESIGNATOR IN DIRECT FIRE        172(1.72%)    189(1.89%)    210(2.10%)    228(2.28%)    257(2.57%)    265(2.65%)
       NUMBER TESTED (% PASSED)       418(58.85%)   428(55.84%)   428(50.93%)   428(46.73%)   428(39.95%)   428(38.08%)

15   ROUND RELIABILITY FAILURE         11(0.11%)      9(0.09%)      8(0.08%)      7(0.07%)      4(0.04%)      5(0.05%)
       NUMBER TESTED (% PASSED)       246(95.53%)   239(96.23%)   218(96.33%)   200(96.50%)   171(97.66%)   163(96.93%)

16   TARGET OBSCURED BY MINI-TERRN     27(0.27%)     31(0.31%)     37(0.37%)     30(0.30%)     19(0.19%)     18(0.18%)
       NUMBER TESTED (% PASSED)       235(88.51%)   230(86.52%)   210(82.38%)   193(84.46%)   167(88.62%)   158(88.61%)

17   ROUND DID NOT ENGAGE TARGET        2(0.02%)      1(0.01%)      0(0.00%)      2(0.02%)      0(0.00%)      2(0.02%)
       NUMBER TESTED (% PASSED)       208(99.04%)   199(99.50%)   173(100.0%)   163(98.77%)   148(100.0%)   140(98.57%)

18   ROUND ENGAGED BUT DID NOT HIT     36(0.36%)     33(0.33%)     24(0.24%)     23(0.23%)     19(0.19%)     21(0.21%)
       NUMBER TESTED (% PASSED)       206(82.52%)   198(83.33%)   173(86.13%)   161(85.71%)   148(87.16%)   138(84.78%)

19   ROUND HIT BUT DID NOT KILL        78(0.78%)     82(0.82%)     88(0.88%)     75(0.75%)     56(0.56%)     66(0.66%)
       NUMBER TESTED (% PASSED)       170(54.12%)   165(50.30%)   149(40.94%)   138(45.65%)   129(56.59%)   117(43.59%)

           ***** KILLS                    92            83            61            63            73            51 *****

       PROB(KILL/DCSN)=                  .92%          .83%          .61%          .63%          .73%          .51%
       PROB(KILL/ATMP ENG)=            16.20%        14.61%        10.74%        11.09%        12.85%         8.98%
       PROB(KILL/SHOT)=                21.50%        19.39%        14.25%        14.72%        17.06%        11.92%
       PROB(LOS)=                      100.0%        100.0%        100.0%        100.0%        100.0%        100.0%
       PROB(SHOT/DCSN)=       4.28%    PROB(SHOT/ATMP ENG)=     75.35%     PROB(ATMP ENG/DCSN)=     5.68%

DESIGNATOR SURVIVABILITY: CASES IN WHICH DESIGNATOR KILLED =   184(1.84%)      NUMBER TESTED (% NOT KILLED) =  418(55.98%)
```

LISTING D-4 SAMPLE OUTPUT FROM COPE

(PAGE 7 OF 11)

```
* * * * * * * * * * * * * * * * * * * * * * * * *   C A S E   N U M B E R   3   * * * * * * * * * * * * * * * * * * * * * * * * *

NUMBER OF REPLICATIONS=10000

                 TARGET

 TYPE  4
 NUMBER OF VEHICLES= 6
 MEAN DISTANCE BETWEEN VEHICLES= 50.M
 TARGET VELOCITY=3. M/SEC
 TARGET HEADING=  0. DEGREES
 TARGET'S REFLECTIVITY= .10

              FIRING INFO

 NUMBER OF ROUNDS PER ENGAGEMENT= 6
 ROUND RELIABILITY= .96
 TIME BETWEEN ROUNDS= 10. SEC
 TIME OF FLIGHT= 31.0 SEC
 NOMINAL GUN-TARGET RANGE=  8. KM
 ANGLE T= 25. DEGREES
 DEFLECTION BIAS=   0. METERS

                 MISC

 DELTA T3 NOT ASSUMED ZERO
 RUMC= 200.M
 SEEKER SENSITIVITY= 60. JOULES/M**2

      DETECT, COMMO, AND PROC TIMES

 PRIORITY PRE-PLANNED TARGET
 DIGITAL COMMO
 DAY TIME
 NOMINAL RESPONSE TIME=106. SEC
 PROB SUCCESSFUL DIGITAL TRAN= .915
 PROB CORRECT MESSAGE =  .975
 PROB DD WARNED TO LASE =  .975

               DESIGNATOR

 TYPE  1
 MAXIMUM DESIGNATOR RANGE=7000. M
 BAIL-OUT RANGE=1000. M
 DESIGNATOR AT VANTAGE POINT

          DESIGNATOR SUPPRESSED

 PROB SUPPRESSED BY INDIRECT FIRE=.01
 PROB SUPPRESSED BY DIRECT FIRE: NONE

                 WEATHER

 MONTH OF DEC
 TIME OF DAY: 2200

             TERRAIN AND LOS

 AVERAGE TERRAIN
 BROKEN LOS ("RANDOM OCCURRENCE")

               OBSCURANTS

 PROB SMOKE KILLS MISSION= .22
 VOLUME OF SMOKE FIRED:  200 +  400
 PROB DUST KILLS MISSION= .37

$$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$      C O M M E N T S      $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$

** USER **  THIS COMMENT WILL BE PRINTED ON CASE 3 HEADING PAGE
** COPE **  CHOICE OF DAY TIME   FOR RESPONSE TIME DATA IS INCONSISTENT WITH TIME OF 2200 FOR WEATHER DATA
** COPE **  TEMPORARY OPTION IN EFFECT FOR RODATA
```

LISTING D-4 SAMPLE OUTPUT FROM COPE

(PAGE 8 OF 11)

| | CAUSE OF ABORT | | | |
|---|---|---|---|---|
| | | ***** OCCASIONS (10000) ***** | | |
| 1 | DESIGNATOR IN INDIRECT FIRE | 84(0.84%) | NUMBER TESTED (% PASSED): | 10000(99.16%) |
| 2 | TARGET BEYOND VISIBILITY RANGE | 1641(16.41%) | NUMBER TESTED (% PASSED): | 9916(83.45%) |
| 3 | TARGET BEYOND DESIGNATOR RANGE | 0(0.00%) | NUMBER TESTED (% PASSED): | 8275(100.0%) |
| 4 | TARGET NOT DETECTED | 0(0.00%) | NUMBER TESTED (% PASSED): | 8275(100.0%) |
| 5 | SMOKE KILLED MISSION | 1853(18.53%) | NUMBER TESTED (% PASSED): | 8275(77.61%) |
| 6 | DUST KILLED MISSION | 2396(23.96%) | NUMBER TESTED (% PASSED): | 6422(62.69%) |
| 7 | DESIGNATOR BAILED-OUT PRE-COMM | 588(5.88%) | NUMBER TESTED (% PASSED): | 4026(85.39%) |
| | | ***** ATTEMPTED ENGAGEMENTS ( 3438) ***** | | |
| 8 | COMMO OUT (BOTH VOICE & DIGTL) | 273(2.73%) | NUMBER TESTED (% PASSED): | 3438(92.06%) |
| 9 | ERROR IN TRANSMISSION | 83(0.83%) | NUMBER TESTED (% PASSED): | 3165(97.38%) |
| 10 | DESIGNATOR BAILED-OUT POST-COM | 237(2.37%) | NUMBER TESTED (% PASSED): | 3082(92.31%) |
| 11 | LOS LOST - NO FIRING | 0(0.00%) | NUMBER TESTED (% PASSED): | 2845(100.0%) |

***** SHOTS ( 2845) *****

| | ROUND NUMBER : | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| 12 | LOS LOST DURING MISSION | 531(5.31%) | 562(5.62%) | 600(6.00%) | 660(6.60%) | 715(7.15%) | 785(7.85%) |
| | NUMBER TESTED (% PASSED) | 2845(81.34%) | 2845(80.25%) | 2845(78.91%) | 2845(76.80%) | 2845(74.87%) | 2845(72.41%) |
| 13 | DESIGNATOR NOT WARNED IN TIME | 60(0.60%) | 0(0.00%) | 0(0.00%) | 0(0.00%) | 0(0.00%) | 0(0.00%) |
| | NUMBER TESTED (% PASSED) | 2314(97.41%) | 2283(100.0%) | 2245(100.0%) | 2185(100.0%) | 2130(100.0%) | 2060(100.0%) |
| 14 | DESIGNATOR IN DIRECT FIRE | 0(0.00%) | 0(0.00%) | 0(0.00%) | 0(0.00%) | 0(0.00%) | 0(0.00%) |
| | NUMBER TESTED (% PASSED) | 2254(100.0%) | 2283(100.0%) | 2245(100.0%) | 2185(100.0%) | 2130(100.0%) | 2060(100.0%) |
| 15 | ROUND RELIABILITY FAILURE | 92(0.92%) | 75(0.75%) | 89(0.89%) | 85(0.85%) | 87(0.87%) | 80(0.80%) |
| | NUMBER TESTED (% PASSED) | 2254(95.92%) | 2283(96.71%) | 2245(96.04%) | 2185(96.11%) | 2130(95.92%) | 2060(96.12%) |
| 16 | TARGET OBSCURED BY MINI-TERRN | 376(3.76%) | 382(3.82%) | 384(3.84%) | 351(3.51%) | 317(3.17%) | 327(3.27%) |
| | NUMBER TESTED (% PASSED) | 2162(82.61%) | 2208(82.70%) | 2156(82.19%) | 2100(83.29%) | 2043(84.48%) | 1980(83.48%) |
| 17 | ROUND DID NOT ENGAGE TARGET | 174(1.74%) | 166(1.66%) | 180(1.80%) | 184(1.84%) | 209(2.09%) | 242(2.42%) |
| | NUMBER TESTED (% PASSED) | 1706(93.06%) | 1826(90.91%) | 1772(89.84%) | 1749(89.48%) | 1726(87.89%) | 1653(85.36%) |
| 18 | ROUND ENGAGED BUT DID NOT HIT | 424(4.24%) | 428(4.28%) | 414(4.14%) | 367(3.67%) | 387(3.87%) | 345(3.45%) |
| | NUMBER TESTED (% PASSED) | 1662(74.49%) | 1660(74.22%) | 1592(73.99%) | 1565(76.55%) | 1517(74.49%) | 1411(75.55%) |
| 19 | ROUND HIT BUT DID NOT KILL | 228(2.28%) | 265(2.65%) | 228(2.28%) | 224(2.24%) | 217(2.17%) | 226(2.26%) |
| | NUMBER TESTED (% PASSED) | 1230(81.50%) | 1232(78.49%) | 1178(80.65%) | 1198(81.30%) | 1130(80.80%) | 1066(78.80%) |
| | ***** KILLS | 1010 | 967 | 950 | 974 | 913 | 840 ***** |
| | PROB(KILL/OCSN)= | 10.10% | 9.67% | 9.50% | 9.74% | 9.13% | 8.40% |
| | PROB(KILL/ATMP ENG)= | 29.38% | 28.13% | 27.63% | 28.33% | 26.56% | 24.43% |
| | PROB(KILL/SHOT)= | 35.50% | 33.99% | 33.39% | 34.24% | 32.09% | 29.53% |
| | PROB(LOS)= | 81.34% | 80.25% | 78.91% | 76.80% | 74.87% | 72.41% |

PROB(SHOT/OCSN)= 28.45% PROB(SHOT/ATMP ENG)= 82.75% PROB(ATMP ENG/OCSN)= 34.38%

DESIGNATOR SURVIVABILITY: CASES IN WHICH DESIGNATOR KILLED - 0(0.00%) NUMBER TESTED (% NOT KILLED) = 2254(100.0%)

LISTING D-4 SAMPLE OUTPUT FROM COPE

(PAGE 9 OF 11)

```
* * * * * * * * * * * * * * * * * * * * * * * * *   C A S E   N U M B E R   4   * * * * * * * * * * * * * * * * * * * * * * * * *

NUMBER OF REPLICATIONS=10000

TARGET
TYPE  1
NUMBER OF VEHICLES=10
MEAN DISTANCE BETWEEN VEHICLES= 61.M
TARGET VELOCITY=5. M/SEC
TARGET HEADING=  0. DEGREES
TARGET'S REFLECTIVITY= .30

FIRING INFO
NUMBER OF ROUNDS PER ENGAGEMENT= 6
ROUND RELIABILITY= .96
TIME BETWEEN ROUNDS= 20. SEC
TIME OF FLIGHT= 31.0 SEC
NOMINAL GUN-TARGET RANGE=  8. KM
ANGLE T= 60. DEGREES
DEFLECTION BIAS=   0. METERS

MISC
DELTA T3 ASSUMED TO BE ZERO
SEEKER SENSITIVITY= 60. JOULES/M**2

DETECT, COMMO, AND PROC TIMES
PRIORITY PRE-PLANNED TARGET
DIGITAL COMMO
DAY TIME
NOMINAL RESPONSE TIME=106. SEC
PROB SUCCESSFUL DIGITAL TRAN= .915
PROB CORRECT MESSAGE =  .975
PROB OO WARNED TO LASE =  .800

DESIGNATOR
TYPE  1
MAXIMUM DESIGNATOR RANGE=7000. M
FAIL-OUT RANGE=1000. M
DESIGNATOR AT VANTAGE POINT

DESIGNATOR SUPPRESSED
PROB SUPPRESSED BY INDIRECT FIRE=.01
PROB SUPPRESSED BY DIRECT FIRE: HIGH

WEATHER
MONTH OF JUN
TIME OF DAY: 1400

TERRAIN AND LOS
AVERAGE TERRAIN
CONTINUOUS LOS ("SHOOTING GALLERY")

OBSCURANTS
PROB SMOKE KILLS MISSION= .15
PROB DUST KILLS MISSION= .20
```

LISTING D-4  SAMPLE OUTPUT FROM COPE

(PAGE 10 OF 11)

```
              CAUSE OF ABORT
                                          ***** OCCASIONS (10000) *****

 1   DESIGNATOR IN INDIRECT FIRE          84(0.84%)    NUMBER TESTED (% PASSED):  10000(99.16%)

 2   TARGET BEYOND VISIBILITY RANGE      116(1.16%)    NUMBER TESTED (% PASSED):   9916(98.83%)

 3   TARGET BEYOND DESIGNATOR RANGE        0(0.00%)    NUMBER TESTED (% PASSED):   9800(100.0%)

 4   TARGET NOT DETECTED                 268(2.68%)    NUMBER TESTED (% PASSED):   9800(97.27%)

 5   SMOKE KILLED MISSION               1440(14.40%)   NUMBER TESTED (% PASSED):   9532(84.89%)

 6   DUST KILLED MISSION                1636(16.36%)   NUMBER TESTED (% PASSED):   8092(79.78%)

 7   DESIGNATOR BAILED-OUT PRE-COMM      758(7.58%)    NUMBER TESTED (% PASSED):   6456(88.26%)

                                     ***** ATTEMPTED ENGAGEMENTS ( 5698) *****

 8   COMMO OUT (BOTH VOICE & DIGTL)      474(4.74%)    NUMBER TESTED (% PASSED):   5698(91.68%)

 9   ERROR IN TRANSMISSION               150(1.50%)    NUMBER TESTED (% PASSED):   5224(97.13%)

10   DESIGNATOR BAILED-OUT POST-COM      558(5.58%)    NUMBER TESTED (% PASSED):   5074(89.00%)

11   LOS LOST - NO FIRING                851(8.51%)    NUMBER TESTED (% PASSED):   4516(81.16%)

                                            ***** SHOTS ( 3665) *****
       ROUND NUMBER :                      1             2             3             4             5             6

12   LOS LOST DURING MISSION           583(5.83%)   1058(10.58%)  1529(15.29%)  2008(20.08%)  2388(23.88%)  2679(26.79%)
       NUMBER TESTED (% PASSED)      3665(84.09%)   3665(71.13%)  3665(58.28%)  3665(45.21%)  3665(34.84%)  3665(26.90%)

13   DESIGNATOR NOT WARNED IN TIME     598(5.98%)      0(0.00%)      0(0.00%)      0(0.00%)      0(0.00%)      0(0.00%)
       NUMBER TESTED (% PASSED)      3082(80.60%)   2607(100.0%)  2136(100.0%)  1657(100.0%)  1277(100.0%)   986(100.0%)

14   DESIGNATOR IN DIRECT FIRE         303(3.03%)    385(3.85%)    357(3.57%)    318(3.18%)    280(2.80%)    219(2.19%)
       NUMBER TESTED (% PASSED)      2484(87.80%)   2607(85.23%)  2136(83.29%)  1657(80.81%)  1277(78.07%)   986(77.79%)

15   ROUND RELIABILITY FAILURE          99(0.99%)     75(0.75%)     82(0.82%)     58(0.58%)     31(0.31%)     26(0.26%)
       NUMBER TESTED (% PASSED)      2181(95.46%)   2222(96.62%)  1779(95.39%)  1339(95.67%)   997(96.89%)   767(96.61%)

16   TARGET OBSCURED BY MINI-TERRN     358(3.58%)    392(3.92%)    303(3.03%)    218(2.18%)    154(1.54%)    137(1.37%)
       NUMBER TESTED (% PASSED)      2082(82.80%)   2147(81.74%)  1697(82.14%)  1281(82.98%)   966(84.06%)   741(81.51%)

17   ROUND DID NOT ENGAGE TARGET         7(0.07%)      6(0.06%)      6(0.06%)      5(0.05%)      5(0.05%)      8(0.08%)
       NUMBER TESTED (% PASSED)      1724(99.59%)   1755(99.66%)  1394(99.57%)  1063(99.53%)   812(99.38%)   604(98.68%)

18   ROUND ENGAGED BUT DID NOT HIT     503(5.03%)    502(5.02%)    399(3.99%)    261(2.61%)    211(2.11%)    158(1.58%)
       NUMBER TESTED (% PASSED)      1717(70.70%)   1749(71.30%)  1388(71.25%)  1058(75.33%)   807(73.85%)   596(73.49%)

19   ROUND HIT BUT DID NOT KILL        636(6.36%)    626(6.76%)    504(5.04%)    384(3.84%)    280(2.80%)    202(2.02%)
       NUMBER TESTED (% PASSED)      1214(47.61%)   1247(49.80%)   989(49.04%)   797(51.82%)   596(53.02%)   438(53.88%)

            ***** KILLS                   578           621           485           413           316           236 *****

       PROB(KILL/OCSN)=                  5.78%         6.21%         4.85%         4.13%         3.16%         2.36%
       PROB(KILL/ATMP ENG)-             10.14%        10.90%         8.51%         7.25%         5.55%         4.14%
       PROB(KILL/SHOT)-                 15.77%        16.94%        13.23%        11.27%         8.62%         6.44%
       PROB(LOS)=                       68.25%        57.73%        47.30%        36.69%        28.28%        21.83%
       PROB(SHOT/OCSN)=      36.65%    PROB(SHOT/ATMP ENG)-     64.32%       PROB(ATMP ENG/OCSN)=     56.98%

DESIGNATOR SURVIVABILITY: CASES IN WHICH DESIGNATOR KILLED -   241(2.41%)      NUMBER TESTED (% NOT KILLED) = 2484(90.30%)
```

LISTING D-4 SAMPLE OUTPUT FROM COPE

(PAGE 11 OF 11)



Next page is blank.

# APPENDIX E

## NOTATIONAL CONVENTIONS

In chapters 4, 5, and 14 of this report, the content of input cards is sometimes displayed with underlining. This underlining serves only to distinguish that which goes on the card itself from the remainder of the text; no actual underlining is punched on any of the input cards themselves.

Items which are displayed in capital letters (with underlining) in the description of a card type are to be copied literally, letter-for-letter onto the input card of that type. Items displayed in lower case letters represent positions on the card at which substitutions are to be made either from a list of allowed alternatives listed (in capital letters) in the descriptive text or by supplying an appropriate numerical value. In the lists of character strings allowed as substitutions for option words, only the underlined (capital) characters are used.

Similary, in chapter 7 where various messages and error prints are described, the capitalized, underlined part is printed as shown (but without the underlining) whereas the lower case part describes or substitutes for text which may vary from occasion to occasion.

Next page is blank.

## APPENDIX F

## GLOSSARY OF ABBREVIATIONS

| | |
|---|---|
| BCS | Battery Computer System |
| CDC | Control Data Corporation |
| COPE | (1) Combat Operational Performance Evaluation<br>(2) COPPERHEAD Operational Performance Evaluation |
| DMD | Digital Message Device |
| D.O., DO | (1) Designator Operator<br>(2) DO as in FORTRAN DO-LOOP |
| DT II | Development Test II |
| FDC | Fire Direction Center |
| GLLD | Ground Laser Location Designator |
| HE | High Explosive |
| IMSL | International Mathematical and Statistical Library |
| LDWSS | Laser Designator Weapon System Simulation |
| LOS | Line-of-Sight |
| LTD | Laser Target Designator |
| MULE | Modular Unit Laser Equipment |
| OT II | Operational Test II |
| PAM | Probability of Acquisition and Maneuver |
| PE | Probability of (Seeker) Engagement |
| PRBLOS | Probability of Line-of-Sight |
| PREPMS | Preprocessor of Mass Storage File |
| TLE | Target Location Error |
| USAFAS | US Army Field Artillery School |



Next page is blank.

## BIBLIOGRAPHY


1. Amoruso, Michael, J., Tice F. DeYoung, Dennis D. Ladd, and Roger D. Schulz, A Comprehensive Digital Flight Simulation of the Cannon Launched Guided Projectile, Rodman Laboratory, Rock Island, IL, January 1977, R-TR-77-007, UNCLASSIFIED report.
This report documents a flight simulation model that serves much the same purpose as the LDWSS model. This work was not used in PAM or COPE, but may be of interest to the researcher in this field.

2. Chernick, Julian A., Moving Target Location Errors for Ground Targets, US Army Materiel Systems Analysis Activity, Aberdeen Proving Ground, MD, September 1980, (to be published).
This report contains an updated version of the target location error modeling of moving targets used in PAM and COPE.

3. Chernick, Julian A., Preliminary Analysis of Extended Range COPPERHEAD Operational Performance (U), Ground Warfare Division Interim Note No. G-85, US Army Materiel Systems Analysis Activity, Aberdeen Proving Ground, MD, January 1980, CONFIDENTIAL report.
This report analyzes the performance of extended range COPPERHEAD using one of the COPE variants (RPVCOPE).

4. Chernick, Julian A., Richard C. Scungio, Michael Starks, Utility of COPPERHEAD with Ground Laser Designation in a European Battlefield Environment (U), *Technical Report No. 257, US Army Materiel Systems* Analysis Activity, Aberdeen Proving Ground, MD, December 1978, CONFIDENTIAL report.
This report includes the first efforts to model the effects of weather on COPPERHEAD performance. It contains an analysis of the utility of COPPERHEAD under European weather conditions using particular terrain scenarios.

5. Control Data Corporation, FORTRAN Extended Version 4 Reference Manual, Revision E, CDC Publication No. 60497800, CDC, Publications and Graphics Division, Sunnyvale, CA, July 1979.
This publication documents the version of FORTRAN IV in which the COPE program and its preprocessors are written. It is of special value to anyone attempting to modify COPE for use on any non-CDC machines.

6. Lewis, C.L., A.G. Nichols, and A.W. Lee, User's Guide for the Phase I Laser Designator/Weapon System Simulation (LDWSS) of the COPPERHEAD Guided Projectile System, Vol. I, Technical Report RG-77-25, US Army Missile Command, Redstone Arsenal, AL, July 1977, UNCLASSIFIED report.
This report documents the LDWSS model which was used to generate the probability of hit (given seeker engages target and round can maneuver to target) numbers used with COPE.

7. Starks, Michael, A Monte Carlo Model for Determining COPPERHEAD Probability of Acquisition and Maneuver (U), US Army Materiel Systems Analysis Activity, Aberdeen Proving Ground, MD, September 1980 (to be published).
This report documents the PAM program and the modeling assumptions underlying it in much greater detail than Chapter 16 of the present report.

8. Cost and Operational Effectiveness Analysis (COPPERHEAD COEA) (U), ACN 18312, US Army Field Artillery School, FT Sill, OK, October 1979, SECRET report.
This report cites the sources of the data used with COPE, includes a section similar to Chapter 2 of the present report, and includes the analysis of COPE results. It also includes, of course, the entire COPPERHEAD COEA write-up.

9. Weaver, Jonathan M., and Lawrence Bowman, Multiple Target Simulation (MUTSI) - A Discrete Monte Carlo Technique That Evaluates the Availability of Multiple Enemy Ground Targets in a GLLD/COPPERHEAD Target of Opportunity Situation, Ground Warfare Division Interim Note G-61, US Army Materiel Systems Analysis Activity, Aberdeen Proving Ground, MD, July 1979, UNCLASSIFIED report.
This report had some influence on the modeling of multi-vehicle targets in PAM and COPE.

DISTRIBUTION LIST

| No. of Copies | Organization |
|---|---|
| 12 | Commander<br>The Defense Technical Information Center<br>ATTN: TCA<br>Cameron Station<br>Alexandria, VA 22314 |
| 1 | Commander<br>US Army Materiel Development & Readiness Command<br>ATTN: DRCCP<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333 |
| 1 | Commander<br>US Army Materiel Development & Readiness Command<br>ATTN: DRCDE-F<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333 |
| 1 | Commander<br>US Army Materiel Development & Readiness Command<br>ATTN: DRCPA-S<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333 |
| 1 | Commander<br>US Army Materiel Development & Readiness Command<br>ATTN: DRCQA<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333 |
| 1 | Commander<br>US Army Materiel Development & Readiness Command<br>ATTN: DRCDE-R<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333 |

| No. of Copies | Organization |
|---|---|
| 1 | Commander<br>US Army Materiel Development & Readiness Command<br>ATTN: DRCDE-D<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333 |
| 1 | Commander<br>US Army Materiel Development & Readiness Command<br>ATTN: DRCBSI-L<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333 |
| 1 | Commander<br>US Army Materiel Development & Readiness Command<br>ATTN: DRCBSI-D<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333 |
| 1 | Commander<br>US Army Materiel Development & Readiness Command<br>ATTN: DRCRE-I<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333 |
| 1 | Commander<br>US Army Armament R&D Command<br>ATTN: DRDAR-LCU-DM<br>Dover, NJ 07801 |
| 1 | Commander<br>US Army Armament R&D Command<br>ATTN: DRDAR-LCU<br>Dover, NJ 07801 |
| 1 | Commander<br>US Army Armament R&D Command<br>ATTN: DRDAR-LCW-E<br>Dover, NJ 07801 |

DISTRIBUTION LIST

| No. of Copies | Organization |
|---|---|
| 1 | Commander<br>US Army Armament R&D Command<br>ATTN: DRDAR-DP<br>Dover, NJ 07801 |
| 1 | Commander<br>US Army Armament R&D Command<br>ATTN: DRDAR-SCS-M<br>Dover, NJ 07801 |
| 1 | Commander<br>US Army Armament R&D Command<br>ATTN: DRDAR-SEA<br>Dover, NJ 07801 |
| 1 | Commander<br>US Army Armament R&D Command<br>ATTN: Technical Library<br>Dover, NJ 07801 |
| 1 | Project Manager<br>Cannon Artillery Weapon Systems<br>ATTN: DRCPM-CAWS<br>Dover, NJ 07801 |
| 1 | Commander<br>Rock Island Arsenal<br>ATTN: Technical Library<br>Rock Island, IL 61299 |
| 1 | Commander<br>US Army Armament Materiel<br>Readiness Command<br>ATTN: DRSAR-PE<br>Rock Island, IL 61299 |
| 1 | Commander<br>Harry Diamond Laboratories<br>ATTN: DELHD-SAB<br>2800 Powder Mill Road<br>Adelphi, MD 20783 |
| 1 | Commander<br>US Army Electronics<br>R&D Command<br>ATTN: DRDEL-SA<br>FT Monmouth, NJ 07703 |
| 1 | Project Manager<br>Army Tactical Data Systems<br>ATTN: DRCPM-TDS<br>FT Monmouth, NJ 07703 |
| 1 | Commander<br>US Army Electronics<br>R&D Command<br>ATTN: DRDEL-AP-OA<br>2800 Powder Mill Road<br>Adelphi, MD 20783 |
| 1 | Director<br>US Army TRADOC Systems<br>Analysis Activity<br>ATTN: ATAA-SL<br>White Sands Msl Rg, NM 88002 |
| 1 | Director<br>US Army TRADOC Systems<br>Analysis Activity<br>ATTN: ATAA-T<br>White Sands Msl Rg, NM 88002 |
| 1 | Director<br>US Army TRADOC Systems<br>Analysis Activity<br>ATTN: ATAA-TFC<br>White Sands Msl Rg, NM 88002 |
| 1 | Commander<br>US Army Missile Command<br>ATTN: DRDMI-C<br>Redstone Arsenal, AL 35809 |

DISTRIBUTION LIST

| No. of Copies | Organization |
|---|---|
| 1 | Commander<br>US Army Missile Command<br>ATTN: DRDMI-TGT<br>Redstone Arsenal, AL 35809 |
| 1 | Commander<br>US Army Missile Command<br>ATTN: DRSMI-DP<br>Redstone Arsenal, AL 35809 |
| 2 | Chief<br>Defense Logistics Studies Information Exchange<br>US Army Logistics Management Center<br>ATTN: DRXMC-D<br>FT Lee, VA 23801 |
| 1 | Commander<br>US Army Concepts Analysis Agency<br>8120 Woodmont Avenue<br>Bethesda, MD 20014 |
| 1 | Reliability Analysis Center<br>ATTN: Mr. I.L. Krulac<br>Griffiss AFB, NY 13441 |
| 1 | Commander<br>US Army Training & Doctrine Comd<br>ATTN: ATCD-TM<br>FT Monroe, VA 23651 |
| 1 | Commander<br>US Army Training & Doctrine Comd<br>ATTN: ATCD-C<br>FT Monroe, VA 23651 |
| 1 | Commander<br>US Army Combined Army Combat Development Activity<br>ATTN: ATCA-CA-A<br>FT Leavenworth, KS 66027 |

| No. of Copies | Organization |
|---|---|
| 5 | Commandant<br>US Army Field Artillery School<br>ATTN: ATSF-CD-AD<br>FT Sill, OK 73503 |
| 1 | Commandant<br>US Army Field Artillery School<br>ATTN: ATSF-CTD<br>FT Sill, OK 73503 |
| 1 | Commandant<br>US Army Armor School Combat & Training Development<br>ATTN: ATZK-CD-MS<br>FT Knox, KY 40121 |
| 1 | Commandant<br>US Army Infantry School<br>ATTN: ATSH-CD-CS<br>FT Benning, GA 31905 |
| 2 | HQDA (DUSA-OR)<br>WASH DC 20301 |
| 1 | HQDA (DAMA)<br>WASH DC 20301 |
| | <u>Aberdeen Proving Ground</u> |
| 1 | Commander, USATECOM<br>ATTN: DRSTE |
| 1 | Commander, USATECOM<br>ATTN: DRSTE CS-A |
| 1 | Director, USABRL |
| 1 | Director, USABRL<br>ATTN: DRDAR-TSB-S (STINFO Branch) |
| 1 | Director, USAHEL<br>Bldg. 520 |



*U.S. GOVERNMENT PRINTING OFFICE: 1981-0-720-037/5630

F

7